9

Z MIKROKOMPUTEREM NA TY

Bajtek

NR INDEKSU 353965
PL ISSN 0860-1674

MIESIĘCZNY DODATEK DO SZTANDARU MŁODYCH NR 9(45) WRZESIEŃ 1989 CENA 500 ZŁ

GALERIA BAJTKA



GAME OVER II

WYŚCIG TRWA

Ani na jeden dzień nie ustaje na świecie wyścig technologiczny. Główną rolę grają w nim Stany Zjednoczone. Japonia i coraz bardziej zintegrowana Europa Zachodnia, ale również Związek Radziecki nie ma zamiaru oddawać pola bez walki.

Nowe możliwości w rozwoju techniki obliczeniowej i elektroniki otwierają badania fundamentalne prowadzone przez uczonych z Uniwersytetu Moskiewskiego — Dmitrija Awierina i Konstantina Lichariewa. Wykazali oni, że bloki pamięci współczesnych maszyn cyfrowych, wykonujących po kilka miliardów operacji na sekundę, mogą być co najmniej tysiąc razy bardziej zintegrowane.

Istota ich koncepcji polega na wykorzystaniu tzw. efektu tunelowego. „Jeśli dwie metalowe płytki zbliżymy odpowiednio blisko do siebie — wyjaśnia doktor fizyki Konstantin Lichariew — to elektrony mogą „przeskakiwać" z jednej płytki na drugą. Nowość polega na tym, że dzięki zastosowaniu odpowiedniego materiału pośredniczącego, powierzchnia przejścia może być mniejsza od jednej setnej części mikrometra kwadratowego — czyli setki razy mniejsza, niż w doświadczeniach przeprowadzonych do tej pory w przodujących centrach naukowo-badawczych świata".

Efekt ten został przewidziany przez radzieckich uczonych 4 lata temu. Jego eksperymentalne potwierdzenie uzyskano obecnie dzięki współpracy z uczonymi szwedzkimi z Uniwersytetu Technologicznego w „Göteborgu — profesorami Tordem Claesonem i Pierem Delsingiem.

Uzyskane rezultaty otwierają szerokie możliwości dla rozwoju elektroniki. W pamięciach współczesnych komputerów rozmieszczone jest nie więcej, iż 10 mln elementów logicznych na jednym centymetrze kwadratowym. Odkrycie radzieckich uczonych pozwala upakować na tej samej powierzchni 10 mld elementów logicznych. Praktyczne zastosowanie tego odkrycia nie ogranicza się tylko do komputerów — uważa Konstantin Lichariew. Wykorzystując efekt tunelowy można m.in. zbudować superczułe przyrządy umożliwiające precyzyjny pomiar bardzo słabych prądów.

Zdaniem radzieckiego uczonego, otwiera to również możliwości rozwoju tak przyszłościowej dziedziny jak elektronika molekularna, w której rolę elementów logicznych pełnić będą molekuły...

Tak, ani na jeden dzień nie ustaje na świecie wyścig technologiczny, ale niestety, z coraz mniejszym udziałem Polski, choć oczywiście nie jest to z pewnością tendencja, której nie można byłoby odwrócić. Zamieścimy zresztą wkrótce na ten temat wywiad z profesorem dr. inż. Andrzejem Jakubowskim, dyrektorem Instytutu Technologii Elektronowej, który jest jednym z tych ludzi w Polsce, którzy starają się nie dopuścić do zepchnięcia naszego kraju do roli technologicznej pustyni. Od nas przecież głównie zależy, czy wokół niewielkich źródeł, jakie w dalszym ciągu na tej pustyni istnieją, uda się tworzyć zielone, kwitnące oazy, poszerzające w szybkim tempie swoje terytorium.

Oczywiście, nie będzie to łatwe. Ale wiele oznak wskazuje na to, że nic już nie będzie w Polsce łatwe. Dotyczy to również możliwości kupienia „Bajtka". Perturbacje na rynku papierowo-poligraficznym powodują, że opóźniają się terminy wydań zwykłych i znika gdzieś papier, który miał być przeznaczony na wydania dodatkowe, m.in. na dwa zeszyty czekającego od dawna na druk „Bajtka — tylko o ATARI". Ale jesteśmy dobrej myśli i wierzymy, że uda się do końca roku wszystkie opóźnienia nadrobić.

Nie wiemy niestety, ile pod koniec roku będzie nasze pismo kosztować, bo wyścig cen jest w stanie wprawić w zdumienie każdego ekonomistę z naszego wydawnictwa. Mamy nadzieję, że będzie to jednak suma, którą uznacie za wartą do wydania!

Waldemar Siwiński

ZA MIESIĄC:

- **Wielka unifikacja — dlaczego CP/M!**
- **TOTAL ECLIPSE — mapa dla koneserów**
- **program kopiujący do stacji TIMEX FDD 3000**
- **reportaż z „Giełdy Bajtka"**
- **Side Kick" na Spectrum**
- **Jak zachować zdrowie, pracując przy komputerze**

ROZSTRZYGNIĘCIE KONKURSU GWIAZDKOWEGO

Nasz tradycyjny Konkurs Gwiazdkowy rozstrzygnięty został 17 maja br. Redakcja otrzymała ponad 3000 prawidłowych odpowiedzi. Rozwiązanie brzmiało „Bajtek — stary przyjaciel w nowym roku.

Losowanie głównej nagrody, komputera Commodore 64, ufundowanej przez amerykańską firmę **Buisness Computer International** nastąpiło przed kamerami **„Teleexpresu"**.

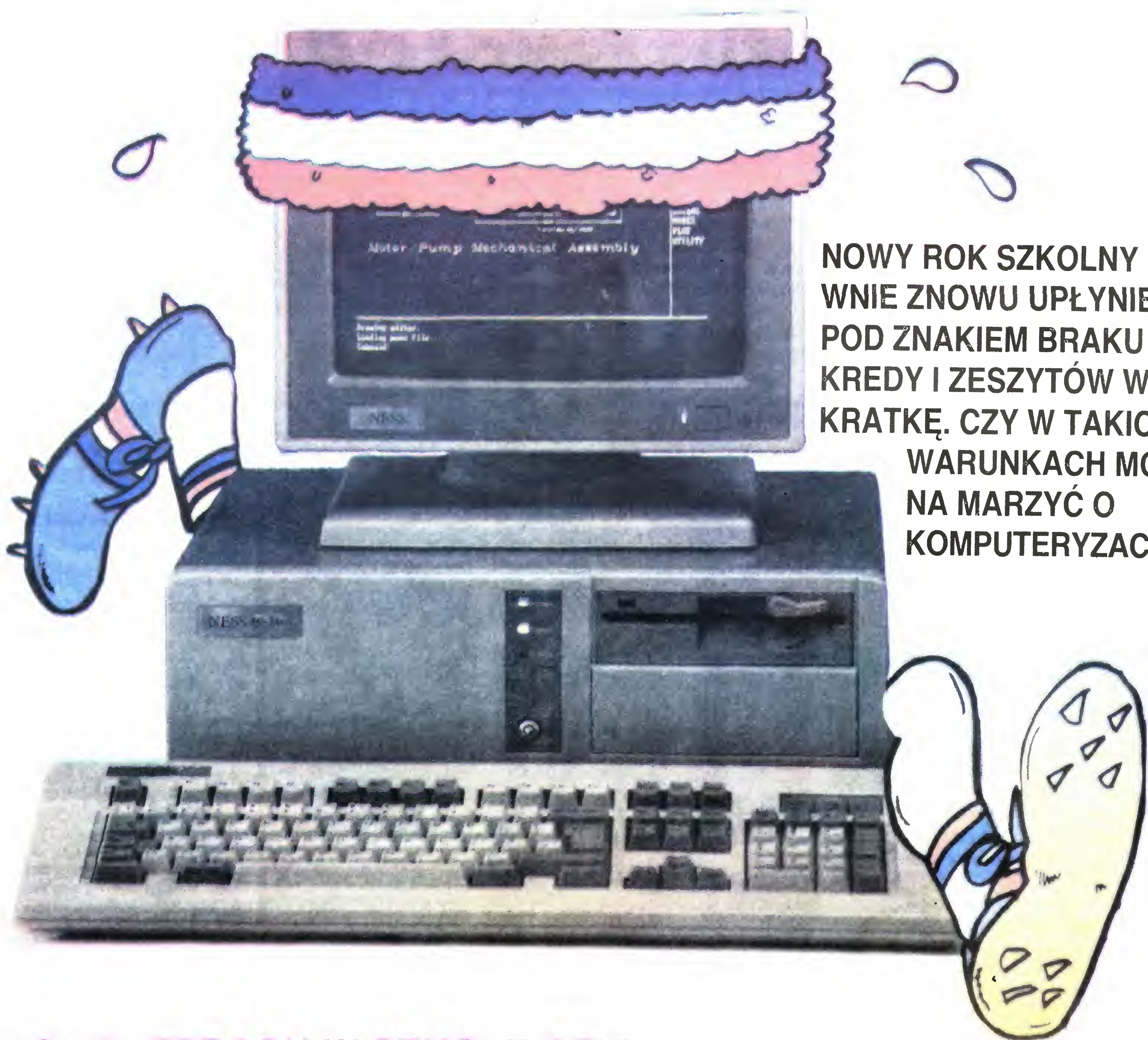
Nagrodę wylosował **Marek Krawczyk z Gdańska**, a informacja o tym podana została na łamach **„Sztandaru Młodych"** i przez **„Teleexpres"** dn 18 maja br.

Bajtek

„BAJTEK" — MIESIĘCZNY DODATEK DO „SZTANDARU MŁODYCH"
ADRES: 00-687 Warszawa, ul. Wspólna 61. Tel. 21-12-05 Przewodniczący Rady Redakcyjnej: Jerzy Domański — redaktor naczelny „Sztandaru Młodych".
ZESPÓŁ REDAKCYJNY: Waldemar Siwiński (z-ca redaktora naczelnego „SM" — kierownik zespołu „Bajtka"), Grzegorz Onichimowski (sekretarz redakcji „Bajtka"), Roman Poznański (kierownik działu klanów), Krzysztof Czernek, Sławomir Gajda (red. techniczny), Andrzej Pilaszek, Sławomir Polak, Wanda Roszkowska (opr. graficzne), Kazimierz Treger, Marcin Waligórski, Roman Wojciechowski. Zdjęcia w numerze: Leopold Dzikowski
klany redagują:
Commodore — Klaudiusz Dybowski, Dominik Falkowski
Amstrad-Schneider — Jonasz Mayer
Spectrum — Marcin Przasnyski
Atari — Wojciech Zientara, Sergiusz Piotrowski

Fotoskład — Tadeusz Olczak,
Montaż offsetowy — Grażyna Ostaszewska,
Korekta — Maria Krajewska, Zofia Wółtańska
WYDAWCA: RSW „Prasa-Książka-Ruch" Młodzieżowa Agencja Wydawnicza, al. Stanów Zjednoczonych 53, 04-028 Warszawa. Telefony: Centrala 13-20-40 do 49, Redakcja Reklamy 13-20-40 do 49 w. 403, 414.
Cena 150 zł.
Skład techniką CRT-200, przygotowalnia offsetowa i druk: PRASOWE ZAKŁADY GRAFICZNE RSW „PRASA-KSIĄŻKA-RUCH" w Ciechanowie, ul. Sienkiewicza 51.
Nr zlecenia 72089 n. 120000 egz. A-111





NOWY ROK SZKOLNY PEWNIE ZNOWU UPŁYNIE POD ZNAKIEM BRAKU KREDY I ZESZYTÓW W KRATKĘ. CZY W TAKICH WARUNKACH MOŻNA MARZYĆ O KOMPUTERYZACJI?

O KOMPUTERACH W SZKOLE CZYLI

POMÓŻ SOBIE SAM!

W Polsce jest ponad 800 tysięcy komputerów, a znakomita większość spośród nich znajduje się w rękach prywatnych. Fakt ten budzi zdumienie zachodnich ekspertów, tym bardziej, że nasz kraj do najbogatszych nie należy.

Chętnie kupujemy sprzęt komputerowy. Mimo kryzysu nic nie zapowiada załamania rynku, a lipcowa decyzja USA o zniesieniu ograniczeń w eksporcie komputerów osobistych do krajów socjalistycznych spowoduje spadek cen i wzmożone zakupy. Wydaje się, że wszystko, co małe i prywatne — kwitnie, a to co państwowe i „społeczne" — coraz bardziej przypomina usychające drzewo.

Tym pochyłym, na które najłatwiej jest skakać stała się w ostatnich latach szkoła.

PIERWSZE PODEJŚCIE

do komputeryzacji szkolnictwa i szerokiego nauczania informatyki w naszym kraju — mówiąc językiem sportowców — zostało spalone. Od roku 1986, kiedy to władze oświatowe przeznaczyły 800 milionów złotych na edukację komputerową, zakupiono dla szkół ponad 11 tysięcy komputerów. Były to głównie Atari, Commodore Spectrum i Timex-y.

„Okrętem flagowym" programu komputeryzacji szkół miał być produkowany we wrocławskich Zakładach Elektronicznych ELWRO — Junior. Trzy lata temu zapowiadano osiągnięcie od 70 do 100 tysięcy egzemplarzy rocznej produkcji w 1990 roku.

Ambitne plany pozostały na papierze, a dyrektor ELWRO Andrzej Musielak przyznał się jedynie do 7 tysięcy Juniorów, które opuściły fabryczne hale w roku ubiegłym. Ministerstwo Edukacji Narodowej nie miało funduszy na większe zakupy, a problemy z dostawcami podzespołów dopełniły reszty.

Jeśli miała to być podstawa edukacji komputerowej, to okazała się ona niezwykle krucha. Juniora zaprojektowano tak, aby miał możliwość wykorzystania bogatego oprogramowania napisanego na Spectrum, a jednocześnie system operacyjny wzorowany był na systemie CP/M firmy Digital Research.

Idea taka, gdyby została zrealizowana pięć lub sześć lat temu, z pewnością byłaby osiągnięciem na skalę światową. Dziś jest już za późno, tym bardziej, że konstrukcja Juniora oparta została o ośmiobitowy mikroprocesor, a te na naszych oczach dożywają swych dni. Dlatego przygnębiająco brzmi wypowiedź dyrektora

GRA O JUTRO

Musielaka udzielona dziennikarzowi „Przeglądu Technicznego", że — właściwie wyparliśmy konkurencję z rynku mikrokomputerów 8-bitowych. To jakbyśmy wyprzedzili Albanię w ilości telefonów na głowę jednego mieszkańca w Europie. Być może dyrektor Musielak nie słyszał o Atari i Commodore, że o Spectrum nie wspomnę? Jestem pewien, że właściciele tych komputerów są tak liczni, że roznieśliby w ręcznym starciu „Juniorowców" w proch.

PIES POGRZEBANY W „KOŚCI"

Klęska Juniora spowodowana została naszym zacofaniem technologicznym. Osiągnięcia Czochralskiego — twórcy najpopularniejszej do tej pory na świecie metody otrzymywania monokryształów półprzewodnikowych, i prof. Ambroziaka — jednego z współtwórców pierwszych na świecie układów scalonych są już chlubną historią.

Polska od roku 1982 produkuje co prawda w niewielkich ilościach mikroprocesory ośmiobitowe, ale nie znalazły one zastosowania w konstrukcji Juniora. Konieczne stały się zakupy u sąsiadów i na Dalekim Wschodzie. Przy braku dewiz przedsięwzięcie pt. Junior stało się mało opłacalne. A to spalone bezpieczniki i źle działająca klawiatura, a to źle funkcjonujące oprogramowanie.

Jeden z moich kilkunastoletnich przyjaciół tak skomentował swą nieudaną próbę „ujarzmienia" wyrobu Elwro — „Jesteśmy sto lat za małpami!"

Być może dlatego w ostatnich miesiącach coraz częściej powtarzana jest opinia o konieczności rezygnacji z Juniora i ... stworzeniu nowego, opartego o szesnastobitowy mikroprocesor komputera klasy IBM PC

DOŁOWANIE BELFRA

Wyobrażam sobie kłopoty nauczyciela, który nie dosyć, że stanie przed komputerem wątpliwej jakości, to jeszcze będzie miał w klasie kilku „gigantów" obrytych w temacie dzięki wcześniejszym zakupom rodziców w „Peweksie". Byłem świadkiem jak trzynastoletni uczeń jednej z warszawskich szkół „zażywał" naszego redakcyjnego kolegę Wojtka Zientanę „w temacie" ATARI. Wojciech Zientara jest na szczęście autorem siedmiu książek o tym komputerze. Jakie ma szanse belfer po kursach?

Takich sytuacji nie da się uniknąć, tym bardziej, że młodzież i rodzice już dawno na własną rękę zaczęli się komputeryzować, nie czekając na ministerialne programy. Nie ma też podręczników do przedmiotu „informatyka", ale załóżmy, że będą. Jednak to nie wszystko. W szkole obok ucznia najważniejszy jest nauczyciel. Nie sądzę aby na taką pracę reflektowało zbyt wielu absolwentów wydziałów elektroniki czy informatyki, którzy działając w spółkach zarabiają znacznie więcej niż nauczyciele. Wobec tego pozostają kursy i „dokształty", ale one nie gwarantują właściwego poziomu nauczania. Koło się zamyka.

Podobnie rzecz się ma z oprogramowaniem. Co prawda nie brakuje na polskim rynku programów użytkowych i edukacyjnych, głównie dzięki „pirackim" praktykom użytkowników i braku ochrony prawnej oprogramowania. Są też programy napisane u nas np. ‚Miasta", „Źródła światła", „Ułamki". Nie wszystkie napisano na Juniora, a komputer bez oprogramowania jest niewiele wart. Przecież nauczyciel sam nie stworzy potrzebnego mu oprogramowania. Jeśli nawet będzie umiał, to nie starczy mu na to czasu między pisaniem konspektów a sprawdzaniem klasówek.

Co się stanie jeśli komputer się zepsuje? We Francji, kraju który znacznie wcześniej wpuścił komputery do klas zauważono, że prawie połowa sprzętu stoi niewykorzystana z powodu awarii! A w ponad połowie szkół średnich wykorzystanie sprzętu nie przekracza trzech godzin miesięcznie na 1 ucznia. Nie sądzę, aby w naszym kraju udało się zorganizować serwis lepszy od francuskiego.

NAJLEPSZY KUMPEL

Współtwórca „Apple", a ostatnio „NeXT-a", Steve Jobs powiedział kiedyś, że chciałby, aby komputer stał się najlepszym kumplem każdego ucznia. Jobsowi udało się zrealizować swoje plany. Komputery „Apple" były obok Commodore używane w amerykańskich szkołach. Jego najnowsze „dziecko" — „NeXT" został zaprojektowany specjalnie dla szkół i uniwersytetów. Możliwości tego komputera są większe od IBM serii PC. W USA i Europie Zachodniej panuje zgodna opinia, że stworzony przez Jobsa sprzęt, dzięki nowatorskim rozwiązaniom technicznym i software'owym stanowi prototyp komputera lat dziewięćdziesiątych.

NeXT znajdzie prawdopodobnie wielu naśladowców i być może wyznaczy standard w swej klasie.

Obok nowatorskich pomysłów hardwarowych i ogromnego postępu w dziedzinie oprogramowania — „przyjazne programy", stosowanie „okien", swobodny dostęp do banków danych za pośrednictwem sieci komputerowych — coraz większą uwagę zwraca się na zastosowania. Nie wystarczy zmusić młodych ludzi od nauki języków programowania, trzeba jeszcze pokazać im, jak wspaniałą pomocą naukową może stać się ich ATARI czy IBM.

Francuskie władze oświatowe realizujące program informatyki dla wszystkich świadomie nie stawiają ostatnio ograniczeń dotyczących zastosowań sprzętu. Komputer może być doskonałym narzędziem dydaktycznym w nauczaniu historii, geografii czy fizyki. Dzięki temu zarówno nauczyciele, jak i uczniowie mają możliwość realizowania własnych pasji i zainteresowań. Oprogramowanie i sprzęt powinno być dostosowane do potrzeb szkoły, jednak nie wolno rozpatrywać tego problemu w oderwaniu od otoczenia.

SPRÓBUJ!

W naszym kraju komputery kojarzą się najczęściej ludziom z błędnie naliczonymi rachunkami za telefon, kolejkami na dworcach przed „skomputeryzowanymi" kasami i niemożnością dokonania przelewu bankowego bo „program się zawiesił".

Wiele wody upłynie w Wiśle zanim doczekamy się łatwego dostępu do komputerów. Nie tylko dla nauczycieli i uczniów, ale dla każdego. Nie ma, jak sądzę w tej chwili najmniejszych szans na realizację wcześniejszych planów. Rok szkolny Anno Domini 1989/90 pewnie znowu upłynie pod znakiem braku kredy i zeszytów w kratkę. Czy w takich warunkach można marzyć o komputeryzacji?

Odpowiadam. Nie tylko można, ale i trzeba! Życie pokazuje nam, jak wiele obecnie zależy od techniki i ludzi z nią związanych. Podziwiamy wspaniałe samochody, magnetowidy, sprzęt radiowy i telewizyjny. Emocjonujemy się wyścigiem bitów i bajtów i coraz powszechniejsze staje się przeświadczenie, że my tego nie potrafimy. Nasza gospodarka nie jest w stanie zapewnić nam powszechnej dostępności elementów elektronicznych, a kultura pracy nie zawsze odpowiada nawet wymogom produkcji łopat. Taki jest stan na dziś.

Ale spróbuj sobie wyobrazić, drogi Czytelniku, że te problemy Ciebie nie dotyczą. Jeśli masz w domu komputer, nawet starą „Spectrumnę", to wiesz, ile z niej można wycisnąć. Tu nie ma lipy. Wszystko zależy od Ciebie, od Twojej wiedzy i umiejętności. Dlatego właśnie nie wpadasz w kompleksy, nie jesteś „do tyłu" wobec Anglika czy Amerykanina. A powiem Ci, że ze swoim najlepszym kumplem możecie zrobić więcej niż oni, bo trzeba się czasem nieźle wysilić aby otrzymać to, co oni po prostu kupują w sklepach. Dlatego przyszłość należy do Ciebie.

Jeśli w swojej szkole trafisz na lekcję z podstaw informatyki, nie znęcaj się nad belfrem, „daruj go zdrowiem" — to też człowiek.

Marek Czarkowski

MICRO



EMULATOR PLOTERA

DO DRUKARKI LASEROWEJ

Rozdzielczość drukarek laserowych 300 punktów na cal jest porównywalna z rozdzielczością typowych ploterów, wynoszącą 0.1 mm. Wykorzystując ten fakt, firma Pacific Data Products oferuje „Plotter in a Cartridge", przystawkę do drukarki laserowej Hewlett-Packarda, która pozwala na emulację języka HPGL (Hewlett-Packard Graphics Language). Efektem jest ponad stukrotnie większa szybkość sporządzania rysunków, przy możliwości zdefiniowania 20 piórek i 48 grubości linii. Mimo ograniczenia rozmiaru strony do formatu A4, wydaje się, że urządzenie to może znaleźć szersze zastosowanie, dzięki istotnemu przyśpieszeniu prac inżynierskich typu CAD/CAM.

CO SŁYCHAĆ U HINDUSÓW?

Indyjscy uczeni postawili przed sobą zadanie organicznego sprzęgnięcia mózgu człowieka z komputerem. Pierwsze doświadczenia, przeprowadzone przez naukowców z oddziału elektroniki Ahmabadskiego Instytutu Naukowo-Badawczego dały zachęcające rezultaty. W charakterze „królika doświadczalnego" wystąpił pilot samolotu, którego głowa została dosłownie „omotana" przewodami i czujnikami, reagującymi na zmiany pól elektrycznych w mózgu i przekazującymi otrzymane sygnały do przyrządów sterowania samolotem. Pośrednikiem między mózgiem i układami sterowania samolotu był system złożony z 64 połączonych ze sobą komputerów.

Przeprowadzając swoje doświadczenia Hindusi wychodzą z założenia, że mózg człowieka jest na tyle doskonały, iż sam dopasuje się do komputera, jeśli tylko stworzony zostanie odpowiedni przetwornik impulsów między nimi.

Warto przy okazji poinformować, że od 1985 roku zainstalowano w Indiach 205 tys. mikrokomputerów, 9,2 tys. komputerów małych i średnich oraz 360 dużych maszyn cyfrowych. W roku bieżącym przemysł indyjski wyprodukuje 75 tys. mikrokomputerów oraz 3500 średnich i dużych maszyn cyfrowych.

(ws)



AT 386 Z ZEGAREM 32,5 MHZ

Najszybszym komputerem klasy IBM PC jest obecnie produkt firmy System Integration Associates, którego sercem jest 32-bitowy procesor Intel 80386, taktowany zegarem o częstotliwości 32.5 MHz. Ze względu na duże ilości ciepła wydzielane przez układ, nominalnie przeznaczony do pracy na 25 MHz, zastosowano dodatkowy wentylator chłodzący. Komputer wyposażono w 4MB pamięci DRAM o czasie dostępu 70ns. Dalsze zwiększenie szybkości działania osiągnięto przez wprowadzenie 64 KB bardzo szybkiej pamięci notatnikowej (cache memory) zrealizowanej na układach typu SRAM o czasie dostępu 15 ns. Maksymalnie pamięć komputera może być rozszerzona do 32 MB. Nietypowo rozwiązano również problem pamięci masowych. Zastosowano specjalizowany kontroler ESDI firmy Adaptec, który pozwala osiągnąć dużą szybkość przesyłania danych — 900 KB/s. Komputer wyposażono w twardy dysk o pojemności 150 MB i czasie dostępu 16.5 ms. Do archiwizacji danych służy streamer o tej samej pojemności. Sterownikiem obrazu jest 8-bitowa karta VGA, pozwalająca na pracę z analogowym monitorem kolorowym. Możliwe jest dodanie koprocesora numerycznego zarówno Intel 80387 jak i Weitek 3167. Pełny zestaw kosztuje w granicach 20 000 $ i oferuje szybkość działania o 30% większą w porównaniu do typowych konstrukcji pracujących z zegarem 25 MHz.

PROGRAM

W naszym kraju przyzwyczailiśmy się płacić duże pieniądze za sprzęt komputerowy i nic za oprogramowanie. Tendencje na rynkach zachodnich są trochę inne. Firma Inovatic sprzedaje za 2495 $ program ReadStar VI, który umożliwia czytanie dokumentów przez komputer IBM PC wyposażony w scanner optyczny. Program rozróżnia dużą ilość czcionek i formatów pisma, zamieniając wczytane teksty na zbiory typu ASCII. Zgodnie z zapewnieniami firmy, dokładność przetwarzania wynosi 99.9%, a szybkość — 4500 znaków na minutę. Cena wymaganego sprzętu stanowi 50—70% ceny programu

ZDALNA KLAWIATURA DO IBM PC

Standardowym wyposażeniem odbiorników telewizji kolorowej jest urządzenie do zdalnego sterowania, pozwalające na zmianę kanałów, siły głosu i parametrów obrazu. Rezultatem przeniesienia tej idei do komputerów osobistych jes zdalna klawiatura do IBM PC, produkowana przez kalifornijską firmę Forte Communications. Za 395 $ otrzymujemy pakiet składający się z ręcznego nadajnika na podczerwień, stacjonarnego odbiornika podłączonego przez interface szeregowy do komputera i dyskietki z oprogramowaniem. 40 klawiszy urządzenia nadawczego pozwala sterować komputerem z odległości nieprzekraczającej 16 m. Podstawowe zastosowanie to działalność dydaktyczna wspomagana komputerowo.



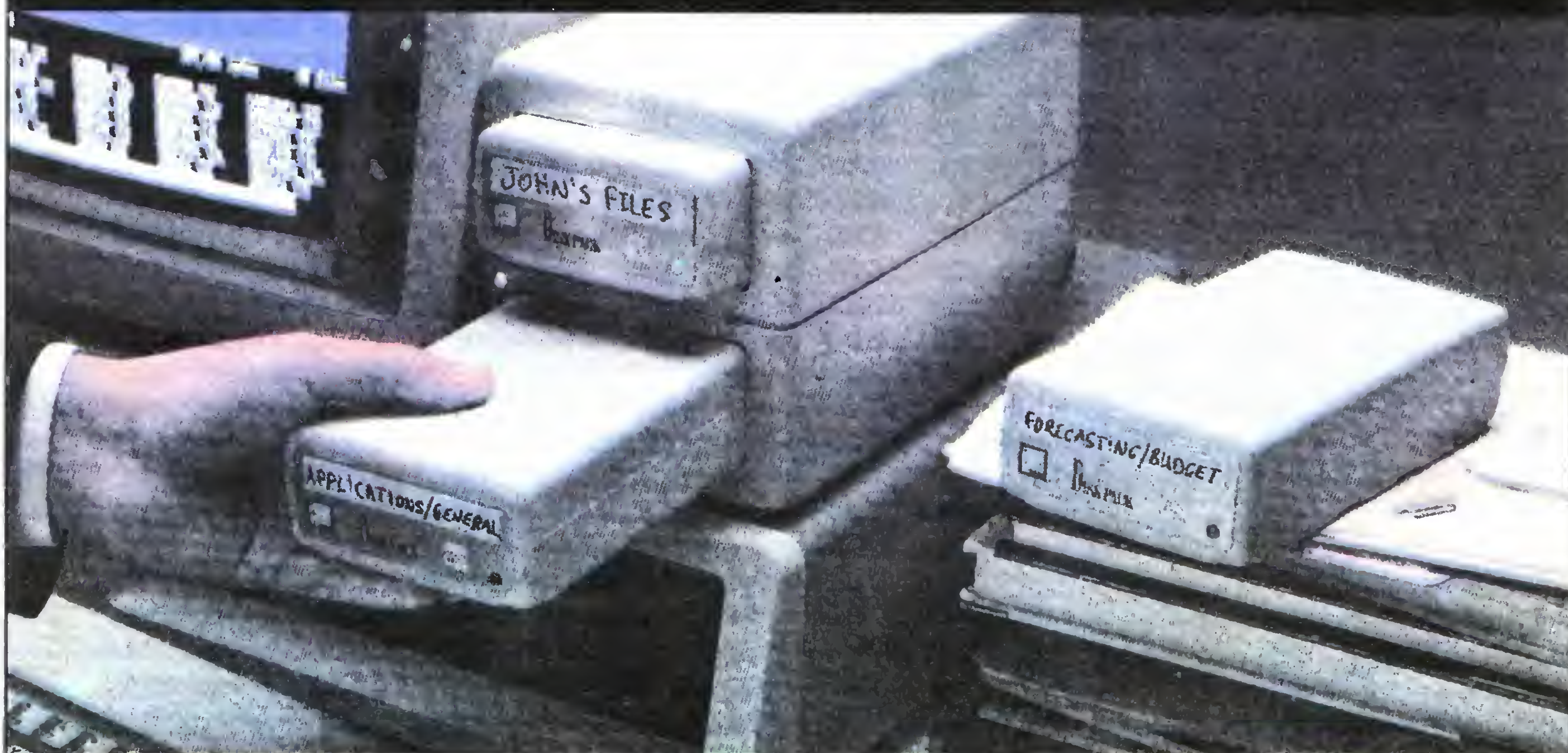
STREAMER 500MB

Streamer jest jednostką pamięci kasetowej przeznaczoną do archiwacji danych z twardego dysku. Typowe pojemności kaset 40—60 MB pozwalają na przechowanie, w wygodny sposób, zawartości standardowego dysku 20—30MB. W przypadku pamięci masowych o większej pojemności konieczne jest stosowanie kilku kaset. Nowością tego rynku jest streamer QFA-500 produkowany przez amerykańską firmę Colorado Memory Systems Incorporation. Dzięki specjalnemu algorytmowi kompresji danych w czasie rzeczywistym, opracowanemu przez firmę Stac of Pasadena, i wydłużeniu taśmy w kasecie z 600 do 1000 stóp, uzyskano urządzenie pozwalające na zapisanie 500 MB informacji na pojedynczej kasecie. Szybkość zapisu wynosi 4—6MB/min, zależnie od rodzaju danych. Montowany wewnętrznie streamer kosztuje 1400$, a jego sterownik, w wersji do IBM PC XT/AT, 150$ (300$ dla PS/2) Cena kasety ponad 40$.

NOWE PAMIĘCI RAM

Obecnie w powszechnym użytku znajdują się układy pamięci RAM o pojemności 1Mbit. Niektóre firmy europejskie i japońskie oferują w niewielkich ilościach kości o czterokrotnie większej pamięci. Celem intensyfikacji prac w tej dziedzinie Texas Instrument i Hitachi podjęły wspólne prace badawcze nad wdrożeniem produkcji układów 16Mbit. Ambitniejszy projekt realizowany jest w japońskiej firmie Fujitsu. Firma ta ogłosiła szczegóły techniczne dotyczące badań nad technologią pozwalającą wytwarzać układy o pojemności 64 Mbit. Osiem takich kości będzie mogło pomieścić ponad 3000 stron standardowego maszynopisu.

WYMIENNY DYSK TWARDY DO IBM PC



Typowe pamięci masowe stosowane w komputerach klasy IBM PC to pięciocalowe napędy dysków elastycznych, dyski twade i streamery. Wymienne dyskietki o pojemności 360 KB lub 1.2 MB służą do przenoszenia zbiorów pomiędzy komputerami. Dysk twardy zapewnia dość dużą, ale niewymienną pamięć rzędu 20—80 MB, natomiast streamer umożliwia archiwowanie niezbędnych informacji na wypadek awarii twardego dysku. Nową ofertę w dziedzinie pamięci masowych stanowi DISK PACK amerykańskiej firmy Mega Drive Systems Incorporation. Jest to wymienny dysk twardy łączący w sobie zalety trzech poprzednio opisanych urządzeń. Charakteryzuje się on dużą pojemnością — od 20 do 120MB, zależnie od wersji, oraz krótkim czasem dostępu rzędu 13 ms. Może być używany do przenoszenia danych między dwoma komputerami, a także do celów archiwacyjnych. Cena kompletnego systemu zawierającego napęd i pakiet dyskowy o pojemności 20 MB wynosi 790$.

Opr. kolumny: JM

CHAOS LOADER

Loader jest to program, który zainstalowany na dyskietce potrafi wyświetlić jej katalog i umożliwia wczytanie dowolnego programu (najczęściej gry), znajdującego się na niej. Znane mi LOADER-y miały bardzo poważne wady, co zmusiło mnie do napisania tego programu.

CHAOS LOADER ma dwie podstawowe zalety, dzięki którym można go ocenić wyżej niż inne jemu podobne programy:

— praktycznie nie zajmuje on miejsca na dyskietce, ponieważ cały mieści się w trzech pierwszych sektorach (BOOT). Można więc zainstalować go na wypełnionej w całości dyskietce (000 FREE SECTORS) nie niszcząc przy tym żadnej gry na niej zapisanej!

— sam rozpoznaje i obsługuje trzy gęstości zapisu: pojedynczą, rozszerzoną i podwójną. Umożliwia to współpracę i pełne wykorzystanie możliwości stacji LDW Super 2000, California Access 2001 oraz stacji z przeróbkami typu Top-Drive lub Happy Warp, które pozwalają na uzyskanie podwójnej gęstości zapisu.

Inne zalety CHAOS LOADER-a to:

— obsługa błędów, dzięki której po błędzie lub przerwaniu ładowania (przez naciśnięcie BREAK lub RESET), komputer nie blokuje się, lecz po przywróceniu standardowego zestawu znaków, kolorów i trybu wyświetlania, następuje wyświetlenie katalogu,

— bardzo krótki czas ładowania,

— ustawienie komórki o nazwie MEMLO, co umożliwia współpracę z programami relokowalnymi.

Ze względu na bardzo małą długość, program ma także dwie wady:

— nie działa ze starym systemem operacyjnym, ponieważ wywołuje procedurę w pamięci ROM przez bezpośredni skok do niej,

— wyświetla jedynie 20 tytułów, co jednak nie jest dużą wadą, ponieważ praktycznie nie zdarzają się dyskietki z taką ilością gier.

CHAOS LOADER można zapisać na dyskietce przy pomocy programu instalującego. Aby go otrzymać należy:

— włączyć stację dysków,

— włożyć do stacji dyskietkę z dowolnym DOS-em,

— włączyć komputer bez wciskania klawisza OPTION,

— wpisać poniższy program,

— włożyć do stacji dyskietkę, na której ma być zapisany program instalujący,

— uruchomić program (RUN).

Błędnie wpisane dane zostaną zasygnalizowane komunikatem. Jeżeli taki komunikat nie wystąpi, to na dyskietce otrzymamy gotowy program o nazwie CHINIT. COM. W celu zainstalowania CHAOS LOADER-a należy wczytać program inicjalizujący przy pomocy dowolnego DOS-a lub LOADER-a, następnie włożyć dyskietkę docelową i nacisnąć klawisz RETURN. Nie ma znaczenia, czy wcześniej na tej dyskietce znajdowały się jakieś gry, gdyż instalowanie loadera nie powoduje ich uszkodzenia. Można również dopisywać gry do dyskietki z uprzednio zainstalowanym loaderem, bez obawy o jego zniszczenie. Jeżeli wcześniej na dyskietce znajdował się inny loader („Bootstrap'85", „Sepped", „JBW Loader" lub podobny), to po użyciu programu instalującego zostanie on zastąpiony CHAOS LOADER-em.

Po włączeniu stacji, włożeniu do niej dyskietki z zainstalowanym CHAOS LOADER-em i zapisanymi grami oraz po włączeniu komputera, na ekranie ukaże się katalog dyskietki, w którym każda nazwa programu będzie oznaczona literą alfabetu. Wyboru programu do załadowania można dokonać przez naciśnięcie klawisza oznaczonego literą odpowiadającą danej pozycji w katalogu. Naciśnięcie klawisza ESC powoduje opuszczenie loadera i przejście do SELFESTU-u komputera. Wciśnięcie innego klawisza powoduje odczytanie nowego katalogu dyskietki, dzięki czemu można wczytać grę z dyskietki bez zainstalowanego loadera (wystarczy wymienić dyskietki).

Życzę przyjemnej pracy z programem CHAOS LOADER.

Janusz Pelc

```
EH  1 REM CHAOS LOADER INSTALATOR
NA  2 REM Janusz Pelc
UQ  3 REM Copyright (c) Bajtek
AY  10 REM
CA  20 OPEN #1,8,0,"D:CHINIT.COM":TRAP 70
PV  30 DIM A$(28):L=100:S=10:RESTORE L
EP  40 READ A$:? CHR$(28);"Wiersz ";L:CS=0
    :FOR X=1 TO LEN(A$)-2 STEP 2:GOSUB 80:
    CS=CS+BYTE:IF CS>255 THEN CS=CS-256
OD  50 PUT #1,BYTE:NEXT X:GOSUB 80:IF CS=B
    YTE THEN L=L+S:GOTO 40
TF  60 ? " popraw!";CHR$(253)
YV  70 END
IM  80 A=ASC(A$(X,X)):B=ASC(A$(X+1,X+1)):B
    YTE=16*(A-48-39*(A>96))+B-48-39*(B>96)
    :RETURN
ED  100 DATA ffff00707f710003000750e420bc
EK  110 DATA 20f4a90a8de802a042a20dbd72fe
RO  120 DATA 089158e003b00cb5589543bd6fa1
BP  130 DATA 0895098d010386468e4402861f7c
VT  140 DATA 88ca10e0a9538d02032053e43057
IM  150 DATA 65a9698545a5458547a90185486e
LR  160 DATA 201b08cabd8008f05029dfc90366
LT  170 DATA f004c942d03aa446bd8308998054
OA  180 DATA 00bd840899a800c014b034b09789
LC  190 DATA e64698a08569219143c8a90e9157
TE  200 DATA 43c8c8bd8508e91f9143e8c09233
DV  210 DATA 90f3a927654385439002e6448a09
OY  220 DATA 691029f0aa10afe645d09f20f8ad
LJ  230 DATA f2c91bd0034c71e4851f295fe95f
XT  240 DATA 41aac546b0b7b5808547b5a88540
TF  250 DATA 48864920120885432012088544 1c
XP  260 DATA 2543c9fff0f0201208854520124 6
MI  270 DATA 088546a91a8de202a9088de3022a
KG  280 DATA d006e643d002e644201208a000d5
YD  290 DATA 9143a544c546d0eda543c545d047
IB  300 DATA e78a4820650868aad0b820680870
CW  310 DATA a51ff0924c74e4e449b005bd8009
KR  320 DATA 08e860a5478d0a030548f0e4a299
CE  330 DATA 03bd6b089d0203ca10f7e8a0012f
KO  340 DATA adea022920d003a280888e0803f8
TG  350 DATA 8c0903a5488d0b032059e430bf6c
KJ  360 DATA ae0803cabd80088549bd7f08855f
JA  370 DATA 47bd7e0829038548a200f0b16c32
QZ  380 DATA e2026ce002525280080106078 0ec
WM  390 DATA a3e8e1eff380acefe1e4e5f28085
LH  400 DATA 8071cc714c8371a9018d0103a952
NP  410 DATA 578d0203a9008d0403a9708d05d1
SJ  420 DATA 03a9018d0a03a9008d0b03ad0139
VU  430 DATA 708dcd712053e4301b18ad0403a9
HY  440 DATA 69808d04039003ee0503ee0a0301
XA  450 DATA d003ee0b03cecd71d0e08c4b73d5
ZG  460 DATA 84ce60ce714b73a90085cda20753
CP  470 DATA f806ce65cd85cdca10f7d8a0019a
ZE  480 DATA 20ea71a5cd4a4a4a4a290f1869ce
FF  490 DATA b09946738860adf40248a9018d0c
VQ  500 DATA f002a200bd30e09d3088bd00e154
ZR  510 DATA 9d0089bd00e29d008abd00e39d29
SM  520 DATA 008bcad0e5a9888df40220ce711d
IN  530 DATA a9788d4403a9728d4503a2afade3
VK  540 DATA 4b7310052056f5a2d38e4803a22e
JZ  550 DATA 008e49038e4b73a90b8d420320cc
EQ  560 DATA 56e4a2ff8efc02ecfc02f0fbade9
VY  570 DATA fc028efc028611c90cd00620806c
NO  580 DATA 714cf871c91cd0e02020f4688de4
XY  590 DATA f402a9008df002607d9b20205026
SS  600 DATA 726f6772616d20696e7374616c33
BR  610 DATA 756a21637920436861 6f73204c56
TD  620 DATA 6f616465729b20202d2d2d2d2dc7
BJ  630 DATA 2d2d2d2d2d2d2d2d2d2d2d2d2d49
BL  640 DATA 2d2d2d2d2d2d2d2d2d2d2d2d2d49
NH  650 DATA 2d9b9b2020576369736b616a2190
OY  660 DATA 6399d2c5d4d5d2ce196d6f24655a
QN  670 DATA 737a207a6170697361259b202095
ZO  680 DATA 6c6f61646572206e6120646f77d0
QL  690 DATA 6f6c6e656a206479736b69657435
EP  700 DATA 63659b9b2020416279206f7075ce
DI  710 DATA 236369252070726f6772616d204c
MT  720 DATA 776369236e696a99c5d3c3199b4f
PP  730 DATA 9b2020a0d7f9f3f4a1f0e9a2a0ee
UD  740 DATA e2a2a1e4a0fae1f0e9f3f5a0eed3
VL  750 DATA f5ede5f2a0b1aeaea09b9b00003c
HP  760 DATA 882f880000000000000000000003f
LB  770 DATA 3c063e663e030038181c38183c1f
EC  780 DATA 0000033e603c067c0018007e0c01
II  790 DATA 18307e0000063c6060603c00e044
GN  800 DATA 02e102f4714a
```

WERYFIKATOR

Dawno temu („Bajtek" 5—6/86) zamieszczono program weryfikujący zbiory na kasecie, również w następnych numerach poruszano ten problem.

Przedstawiane rozwiązania miały pewne wady, takie jak:
- zbyt duża objętość weryfikatora, długi czas wczytywania z kasety;
- konieczność zarezerwowania dla nich pewnej ilości wierszy we własnym programie w Basicu;
- procedura rezydowała na dość często wykorzystywanej 6 stronie pamięci.

Mój program pozbawiony jest wyżej wymienionych wad. Zasadniczo weryfikacja realizowana jest przez krótką (tylko 57 bajtów) procedurę maszynową. Program umożliwiający wygodne korzystanie z niej zajmuje tylko jeden wiersz, którego numer może dowolnie zadeklarować użytkownik. Procedura maszynowa jest całkowicie relokowalna, dla wygody umieściłem ją w zmiennej A$, sposób jej użycia jest dowolny, decyduje o nim użytkownik. Procedura weryfikująca wykorzystuje bufor magnetofonu (GASBUFFER) nie może więc uszkodzić danych znajdujących się w pamięci.

Aby uzyskać wygodny w użyciu program 2 (jeden wiersz) należy wpisać program 1, uruchomić go, a otrzymany wiersz (jego numer może być dowolny) zapisać na taśmie instrukcją ENTER "C:". Program wywołujemy skokiem do wiersza (GOTO). Procedura maszynowa wywoływana jest instrukcją: X=USR(ADR(A$),A), jako parametr A należy podać długość przerw: O — długie, 128 — krótkie.

Wielu Czytelników może być zainteresowanych w jaki sposób działa procedura maszynowa i skąd bierze się jej niewielka długość. Listing asemblera opatrzony jest komentarzami, ograniczę się więc tylko do wyjaśnienia jednej kwestii. Mój program wykorzystuje bezpośrednio procedury systemu operacyjnego i IOCB na stronie 0. Procedury te to:
- CASOPEN otwiera magnetofon;
- CASREADBL odczytuje rekord i umieszcza go w buforze magnetofonu;
- CASCLOSE zamyka magnetofon.

Najważniejszą z nich jest CASREADBL. Umieszcza ona pod adresem 1023 wartość oznaczającą typ ostatnio przeczytanego bloku. Możliwe są trzy przypadki:
1. $FE=254 — EOF — rekord kończący plik, 128 zer.
2. $FC=252 — blok składa się ze 128 znaczących bajtów.
3. $FA=250 — blok składa się z mniej niż 128 znaczących bajtów, ich dokładna liczba znajduje się w 128 bajcie.

Użytkownikom Atari zainteresowanym dokładnym poznaniem procedur obsługi magnetofonu znajdującym się w systemie operacyjnym (zajęcie szalenie pasjonujące i czasochłonne) polecam książkę „Mapa pamięci Atari XL/XE. Procedury wejścia/wyjścia" oraz dużo wytrwałości we własnych próbach... Wyniki mogą być zaskakujące np. rekord EOF nie musi składać się z samych zer!

Andrzej Zalewski

LISTING 1

```
YJ  0 REM WERYFIKATOR
LT  1 REM Andrzej Zalewski
UP  2 REM Copyright (c) Bajtek
NI  3 REM
EY  10 GRAPHICS 0:POKE 82,0
HT  20 POSITION 0,2:? "NEW":POSITION 0,20:
    ? "POKE 842,12"
KP  50 POSITION 0,10:? "31999 CLR: DIM A$(
    57):A$=";CHR$(34);
JN  60 FOR N=1 TO 57:READ A:? CHR$(27);CHR
    $(A);:NEXT N:? CHR$(34);":X=USR(ADR(A$
    ),128):END"
HF  70 POSITION 0,0:POKE 842,13:END
XS  100 DATA 104,104,104,133,43,169,4,133,
    42,141,255,3,32,230,252,32,141,253,16,
    251,173,255,3,201,254,208,13,169,47
PF  110 DATA 160,0,145,88,200,169,43,145,8
    8,208,11,169,37,160,0,145,88,169,50,20
    0,145,88,32,207,253,76,2,243
```

LISTING 2

```
SL  31999 CLR :DIM A$(57):A$="hhh[illegible]
    [illegible]
    [illegible]":X=USR(ADR(A$),128):END
```

LISTING 3

```
0100 ; Super weryfikator
0110 ; Andrzej Zalewski
0120 ; Copyright (c) Bajtek
0130 ;
0140 CASOPEN = $FCE6
0150 CASREADBL = $FD8D
0160 CASCLOSE = $FDCF
0170 ;
0180      *=  $0600
0190 ;
0200      PLA
0210      PLA
0220      PLA
0230      STA 43        ; RODZAJ PRZERW
0240      LDA #4        ; PARAMETR DLA CASOPEN
0250      STA 42        ; ZAPAMIETANIE PARAMETRU
0260      STA 1023      ; OCZYSZCZENIE TYPU REKORDU
0270      JSR CASOPEN   ;OTWORZ MAGNETOFON
0280 PO   JSR CASREADBL ;PRZECZYTAJ BLOK
0290      BPL PO        ; CZY BLAD
0300 ERROR LDA 1023     ; WCZYTAJ TYP BLOKU
0310      CMP #$FE      ; CZY EOF
0320      BNE BLAD      ; NIE EOF
0330      LDA #$2F      ; NAPISZ OK
0340      LDY #0
0350      STA (88),Y
0360      INY
0370      LDA #$2B
0380      STA (88),Y
0390      BNE EXIT
0400 BLAD LDA #37       ; NAPISZ ER
0410      LDY #0
0420      STA (88),Y
0430      LDA #50
0440      INY
0450      STA (88),Y
0460 EXIT JMP CASCLOSE
```

ODZYSKIWANIE PROGRAMÓW

Posiadam komputer Atari 65 XE. Jako pamięć zewnętrzną używam radiomagnetofonu RM 221, który jest podłączony z komputerem przez interfejs. Z zapisem i odczytem nie mam żadnych kłopotów. Od czasu do czasu jednak podczas wczytywania programów, zwłaszcza z kaset, których używam od ładnych paru lat, pojawia się błąd nr. 143, który uniemożliwia wczytanie programu. Błąd ten często jest związany z zagięciami samej taśmy magnetofonowej, których podczas intensywnego użytkowania kasety trudno jest uniknąć.

Z myślą o tych wszystkich Czytelnikach „Bajtka", którym podczas wczytywania programów w Basicu pojawia się napis ERROR 143, napisałem program ingerujący w przerwanie odczytu z szyny szeregowej ISRSIR. Ingerencja polegała na tym, że pominąłem fragment porównujący bajt sumy kontrolnej z zawartością rejestru CHKSUM.

Tak zmieniona procedura umieszczona jest na stronie szóstej. Instrukcje POKE w wierszu 60 informują teraz system operacyjny o lokalizacji nowego programu przerwania odczytu poprzez zmianę wektora VSERIN ($020A) tak, aby teraz pokazywał zmieniony program. Teraz wszelkiego rodzaju procedury odczytu zachodzą bez badania poprawności sumy kontrolnej. Umożliwia to wczytanie programów, które poprzednio sygnalizowały błąd 143. Wczytany program można już wylistować oraz spróbować go naprawić. Myślę, że jest to rozwiązanie dużo lepsze, niż pogodzenie się z utratą programu na zawsze. Życzę udanych prób odzyskiwania programów.

Uwaga: każdorazowo po naciśnięciu klawisza RESET należy powtórzyć instrukcje POKE z wiersza 60.

Artur Lisowski

```
QX  10 REM ODZYSKIWANIE PROGRAMOW
FK  20 REM ARTUR LISOWSKI
MC  30 REM COPYRIGHT (C) BAJTEK
BB  40 REM
CV  50 S=0:FOR Q=0 TO 86:READ A:POKE 1536+
    Q,A:S=S+A:NEXT Q:IF S<>10315 THEN ? "B
    LAD W DATA":END
OA  60 POKE 522,0:POKE 523,6:END
OZ  70 DATA 152,72,173,15,210,141,10,210,4
    8,4,160,140,132,48,41,32,208,4,160,142
    ,132,48,165,56,240,15,173,13,210
NR  80 DATA 197,49,240,0,169,255,133,57,10
    4,168,104,64,173,13,210,160,0,145,50,2
    4,101,49,105,0,133,49,230,50,208
SO  90 DATA 2,230,51,165,50,197,52,165,51,
    229,53,144,222,165,60,240,6,169,0,240,
    210,133,60,169,255,133,56,208,206
```

MONITORY ML cz.8

Prawdopodobnie większość Czytelników oczekiwała tego odcinka „Monitorów" w siódmym numerze BAJTKA; niestety z przyczyn niezależnych ode mnie nie byłem go w stanie przygotować na czas, za co serdecznie Czytelników przepraszam.

Dziś pozostaniemy jeszcze przy niepublikowanych instrukcjach mikroprocesorów rodziny 65XX. Zestawienie to obejmowało kilkanaście rozkazów o działaniu dość złożonym i właśnie ich omówienie będzie tematem dzisiejszego odcinka.

W tabeli użyto następujących skrótów:

(,X)	— adresowanie indeksowe pośrednie
(),Y	— adresowanie pośrednie indeksowe
ZP	— adresowanie strony zerowej
ZP,X	— adresowanie strony zerowej indeksowe X
ZP,Y	— adresowanie strony zerowej indeksowe Y
ABS	— adresowanie absolutne
ABS,X	— adresowanie absolutne indeksowe X
ABS,Y	— adresowanie absolutne indeksowe Y
IMDT	— adresowanie natychmiastowe

Pozostaje jeszcze jedna kwestia — nazewnictwo. Otóż niepublikowane rozkazy mikroprocesora mają również swoje trzyliterowe mnemoniki. Ich składnia jest zróżnicowana (w zależności od źródła publikacji) lecz przyjął się pewien standard — pierwsza i ostatnia litera pierwszego rozkazu oraz pierwsza litera drugiego. Np. zestaw rozkazów ASL:ORA oznaczany jest zwykle jako ASO.

ASO (ASL:ORA)
Najpierw wykonywana jest operacja ASL, a następnie operacja logiczna ORA wyniku z zawartością akumulatora.

RLA (ROL:AND)
Najpierw wykonywana jest operacja ROL, a następnie operacja logiczna AND wyniku z zawartością akumulatora.

LSE lub LRE (LSR:EOR)
Wykonuje najpierw operację LSR, a następnie EOR wyniku z zawartością akumulatora.

RRA (ROR:ADC)
Najpierw wykonuje operację ROR, a następnie dodaje do wyniku zawartość akumulatora.

AXS lub SXS (STX:STA)
Zapisuje do danej komórki pamięci wynik operacji logicznej AND pomiędzy akumulatorem i rejestrem X. Operacja AND jest wykonywana pierwsza, dopiero potem następuje zapis.

LAX (LDA:LDX) lub (LDX:LDA)
Powoduje jednoczesne wczytanie do akumulatora i rejestru X tej samej wartości.

DCM lub DCC (DEC:CMP)
Zmniejsza zawartość komórki pamięci o 1 i porównuje z zawartością akumulatora.

INS (INC:SBC)
Zwiększa zawartość komórki pamięci o jeden i odejmuje rezultat od zawartości akumulatora.

ALR lub ADL (AND:LSR)
Wykonuje operację logiczną AND pomiędzy akumulatorem i daną, a wynik poddaje operacji LSR.

ARR lub ADR (AND:ROR)
Tak jak ALR z tym, że na wyniku jest wykonywana operacja ROR.

OAL lub OAT (ORA:AND:TAX)
Najpierw poddaje operacji logicznej ORA zawartość akumulatora (zawsze z liczbą $EE). Następnie poddaje otrzymany wynik operacji logicznej AND z daną i przenosi zawartość akumulatora (czyli rezultat końcowy) do rejestru X.

SAX lub SCA (SBC:AND:STX)
Najpierw przeprowadza operację logiczną AND pomiędzy zawartością akumulatora i rejestru X. Następnie odejmuje od wyniku daną i rezultat całej operacji wpisuje do rejestru X.

TAA (TXA:AND#)
Przenosi zawartość rejestru X do akumulatora i następnie przeprowadza na niej operację logiczną AND z podaną wartością.

A oto tabela kodów powyższych rozkazów:

Kod	ABS	ABS,X	ABS,Y	ZP	ZP,X	ZP,Y	(,X)	(),Y	IMDT
ASO	0F	1F	1B	07	17		03	13	0B
RLA	2F	3F	3B	27	37		23	33	2B
LSE	4F	5F	5B	47	57		43	53	
RRA	6F	7F	7B	67	77		63	73	
AXS	8F			87		97	83		
LAX	AF		BF	A7	B7		A3	B3	
DCM	CF	DF	DB	C7	D7		C3	D3	
INS	EF	FF	FB	E7	F7		E3	F3	
ALR									4B
ARR									7B
OAL									AB
SAX									CB
TAA									8B

Powyższe rozkazy były już częściowo omówione w poprzedniej części „Monitorów". Kody opisane poniżej stanowią ostatnią grupę rozkazów, o działaniu znacznie bardziej złożonym.

Kod rozkazu: $6B (107)
Gdy znacznik trybu dziesiętnego jest równy zero przeprowadza operację AND pomiędzy akumulatorem i podaną wartością. Następnie bit 0 liczby w akumulatorze jest kopiowany do znacznika przeniesienia rejestru słowa stanu. Teraz wykonywana jest operacja logiczna EOR pomiędzy bitami 5 i 6 rejestru słowa stanu, a wynik umieszczany w znaczniku N tego rejestru.

Kod rozkazu: $93 (147)
Wykonywana jest operacja AND pomiędzy akumulatorem i rejestrem X. Do wyniku dodawana jest liczba 1 i całość staje się adresem w trybie adresowania pośrednim indeksowym.

Kod rozkazu: $97 (151)
Wykonywana jest operacja AND pomiędzy akumulatorem i rejestrem X, a wynik staje się adresem w trybie adresowania absolutnym indeksowym (ABS,X).

Kod rozkazu: $9B (155)
Jest to jeden z najbardziej złożonych rozkazów. Najpierw wykonywana jest operacja logiczna AND pomiędzy akumulatorem i zawartością rejestru X przeniesioną do wskaźnika stosu. Następnie operacja AND jest powtarzana pomiędzy wskaźnikiem stosu i starszym bajtem adresu (HI) zwiększonym o 1. Wynik daje nam konkretny adres skoku (LOHI,Y), gdzie LO oznacza młodszy bajt adresu.

Kod rozkazu: $9C (156)
Wykonywana jest operacja logiczna AND pomiędzy zawartością rejestru Y i liczbą określającą starszy bajt adresu (HI) zwiększoną o 1. Określony adres jest wykorzystany w trybie adresowym absolutnym indeksowym (HILO,X).

Kod rozkazu: $9E (158)
Operacja AND jest przeprowadzana pomiędzy rejestrem X i dowolną wartością zwiększoną o 1. Adres jest następnie wykorzystany w trybie adresowym absolutnym indeksowym (HILO,Y).

Kod rozkazu: $9F (159)
Dwustopniowa operacja AND pomiędzy akumulatorem i rejestrem X (pierwsza fazą) a następnie pomiędzy otrzymanym w pierwszej fazie wynikiem i starszym bajtem adresu zwiększonym o 1. Adres ten jest następnie wykorzystywany w trybie adresowania absolutnym indeksowym (HILO,Y).

Kod rozkazu: $BB (187)
Zawartość komórki o adresie HILO,Y poddawana jest operacji AND ze wskaźnikiem stosu przeniesionym do rejestru X. Dalej wykonywane są kolejno rozkazy TXS:TXA.

Na zakończenie dzisiejszego odcinka pragnąłbym wyjaśnić jeszcze kwestię sporną dotyczącą rozkazu NOP w wersji dwu- i trzybajtowej. Rozkazów tych jak wynika z tabeli jest stosunkowo dużo; czy jednak wszystkie z nich to prawdziwe NOP?

Różne źródła wypowiadają się różnie na ten temat. Te najbardziej (moim zdaniem) miarodajne dzielą te rozkazy aż na trzy kategorie, przypisując im przy okazji trzy różne mnemoniki:

NOP : $1A, 3A, 5A, 7A, DA, FA
SKB : $80, 82, C2, E2, 04, 14, 34, 44, 54, 64, 74, D4, F4
SKW : $0C, 1C, 3C, 5C, 7C, DC, FC

Cóż oznaczają mnemoniki SKB i SKW? Są to skróty od SKip Byte i SKip Word (dosłownie: przeskocz bajt i przeskocz słowo). W tym wypadku „słowo" oznacza wyrażenie złożone z dwóch bajtów. Dla większości Czytelników istotny będzie jednak przede wszystkim fakt, iż SKB i SKW powodują po prostu zignorowanie jednego lub dwóch następujących po sobie bajtów w programie; nie jest to całkowicie równoznaczne z rozkazem NOP powodującym wstrzymanie pracy procesora na dwa cykle zegarowe i dopiero potem (jeśli jest to NOP dwu- lub trzybajtowy) „przeskoczenie" określonej ilości bajtów.

Stosując niepublikowane instrukcje mikroprocesorów należy uwzględnić także pewne niebezpieczeństwo za tym się kryjące. Otóż nie każdy mikroprocesor MUSI być w omawiane tu kody wyposażone, nie każdy rozkaz również musi działać dokładnie w opisany tu sposób. Znane są wypadki, że dana instrukcja działała bez zarzutu np. na starszych wersjach (lub seriach) mikroprocesorów nie chciała natomiast działać na nowszych. Niektóre instrukcje mogą też powodować totalne zablokowanie komputera stąd proponowałbym zawsze zapisywać program przed jego uruchomieniem.

I w ten sposób drogi Czytelniku zaczynamy zbliżać się powoli do końca cyklu artykułów poświęconych monitorom języka maszynowego. W następnym odcinku zajmę się omówieniem ich rozszerzeń — tzw. makroasemblerów w skład których wchodzą także monitory. Omówimy trzy takie programy: CBM MACROASSEMBLER DEVELOPMENT SYSTEM (C-64), MERLIN (C-64) oraz świetny program o nazwie LADS dla Commodore 128. Programy te są niestety przeznaczone wyłącznie dla posiadaczy stacji dysków stąd też postaram się także omówić tzw. HIFI ASSEMBLER działający z Commodore 64 i magnetofonem.

Literatura:

Gelfand Felt, Strauch Krsnik
DAS ANTI-CRACKER BUCH
DATA BECKER, 1988
ISBN N 3-89011-253-6

COMPUTE! 5/1987
ISSN 0194-357X

Klaudiusz Dybowski



Drodzy Czytelnicy!

Od września bieżącego roku Klub Komputerowy „MANIAK" rozszerza swą działalność o Amigę. Zakres działalności Klubu będzie obejmował wymianę oprogramowania, doświadczeń oraz literatury. Zamierzamy również (o ile warunki na to pozwolą) zorganizować cykl wykładów poświęconych temu komputerowi. Wszystkich zainteresowanych zapraszamy do Klubu osobiście (Warszawa, Wasilkowskiego 7) lub prosimy o kontakt telefoniczny (40-62-64) po 15 września 1989.

Arkadiusz Jaźwiński, Andrzej Hołowacz, Paweł Bąkowski

DAS ANTI-CRACKER BUCH

Czy myślałeś choć raz o tym w jaki sposób zabezpieczyć swoje programy w sposób rzeczywiście pewny przed ich kopiowaniem? Jeśli tak, to książka, którą zamierzam tu przedstawić jest właśnie dla Ciebie.

Książka o której mowa to DAS ANTI-CRACKER BUCH autorstwa Gelfanda Felta i Straucha Krsnika, wydana w zeszłym roku w RFN ciesząca się do dziś ogromnym wzięciem. Wydawcą książki jest znana nam już firma DATA BECKER z Düsseldorfu.

Problematyka zabezpieczenia programów ma szczególny oddźwięk w Polsce będącej co tu dużo ukrywać istnym komputerowym El Dorado. Ochrona prawna programów nie istnieje wogóle, literatura jest masowo powielana i tłumaczona bez jakiejkolwiek reakcji ze strony odpowiednich instytucji, pomimo, że olbrzymia większość tych publikacji nosi na sobie znaczek COPYRIGHT. Niejednemu z autorów składano na giełdach propozycje zakupu jego własnego programu; dlatego też uważam, że tematyka omawianej książki jest jak najbardziej na czasie.

DAS ANTI-CRACKER BUCH jest publikacją przeznaczoną i dla zaawansowanych i dla początkujących. W 12 rozdziałach autorzy wprowadzają Czytelnika w tajniki stosowania wielu zabezpieczeń — od tych najprostszych do bardzo skomplikowanych. Omówiono tu zabezpieczenia stosowane w BASIC, kodowanie programów za pomocą instrukcji logicznej EOR, tworzenie programów samouruchomiających się, zmianę wektorów, stosowanie niepublikowanych instrukcji mikroprocesora, blokady katalogu dyskietki czy magnetofonu. Czytelnik jest wprowadzany w coraz bardziej skomplikowane techniki zabezpieczania do stosowania specjalnych przystawek znanych pod nazwą „dongle" — zalanych w masie plastycznej układów przyłączanych następnie np. do portu joysticka.

Szczególnie dużo miejsca autorzy poświęcili zabezpieczaniu programów zapisanych na dyskietkach. Omówione tu zostały wszystkie podstawowe techniki takie jak nielegalne ścieżki, wytwarzanie i obsługa celowo tworzonych na dyskietce błędów, zmiana formatu dyskietki, podwójne ścieżki i półścieżki, maskowanie znaków synchronizacji, „killer tracks" itp. Każda z technik jest poparta odpowiednio przygotowanym przykładem w postaci programu (zarówno źródłowego jak i w postaci programu BASIC gotowego do wpisania i uruchomienia). Autorzy nie zapomnieli również o wirusach i piratach elektronicznych w postaci specjalnych modułów — opisano tu kilka technik wykrywania takich kart.

W momencie oddawania książki do druku wirusy nie były jeszcze tak popularne jak w chwili obecnej, stąd też autorzy poświęcili im bardzo mało miejsca. Z drugiej strony jednak, nawet najlepszy wirus w komputerze pozbawionym dysku twardego nie ma zbyt długiego życia choć również ma pewne szanse na dokonanie spustoszenia.

Książka jest przeznaczona przede wszystkim dla Commodore 64 i 128 (zawiera ona specjalny załącznik w którym znajdują się odnośniki do omawianych technik dotyczące ich wykorzystania w Commodore 128). Tym niemniej 80% materiału można z powodzeniem wykorzystać na dowolnym 8-bitowym Commodore — w końcu zabezpieczenie np. dyskietek odnoszą się w przeważającej części do systemu operacyjnego stacji dysków. Niewątpliwie przydatną publikacją uzupełniającą będzie mapa pamięci dla danego komputera.

Ze swej strony gorąco polecam DAS ANTI-CRACKER BUCH wszystkim programistom zainteresowanym zabezpieczaniem swojej często wielogodzinnej pracy przed rączkami handlarzy; kilka z opisywanych tu technik zabezpieczania dyskietek jest w stanie rozłożyć na cztery łopatki nawet najlepsze programy kopiujące.

Książkę może wykorzystać użytkownik o nawet bardzo małym zakresie wiedzy dzięki prawie setce gotowych do użytku programów.

Na zakończenie mała dygresja. Jeden z użytkowników po przeczytaniu tej książki zgłaszał zastrzeżenie, że nie opisuje ona dwóch bardzo złożonych systemów zabezpieczania programów znanych pod nazwami RAPIDLOK i PROLOK. Szanowny Panie, systemy te mają swój własny znak handlowy (TM) i są chronione mniej więcej tak samo jak procedury BIOS i BDOS opracowane przez firmę IBM czy programy firmy MICROSOFT.

Klaudiusz Dybowski

Gelfand Felt, Strauch Krsnik
DAS ANTI-CRACKER BUCH
DATA BECKER GmbH
Rok wydania 1988, stron 379
Cena 39 DM.
ISBN 3-89011-253-6

ZERO PAGE

„Zero page" to angielska nazwa tak zwanej strony zerowej komputera (adresy $00 do $FF). Wszystkie podręczniki podają, że jest ona bardzo „ciężko" wykorzystywana przez system operacyjny komputera.

Niedowiarkom proponuję wpisanie i uruchomienie jednego z podanych tu programów. Następnie trzeba przesunąć kursor w widoczne miejsce i wczytać lub wpisac dowolny inny program (byle nie graficzny. Miejsca na ekranie, w których zmieniają się znaki, to komórki pamięci strony zerowej, w których system operacyjny wykonuje pewne działania ściśle związane z Twoim programem. Zwróć uwagę, że podczas zapisu bądź odczytu programu cała strona „zamiera"; w rzeczywistości jednak jest ona wykorzystywana dalej, lecz przerwania IRQ, w których umieściłem procedurę, są „podkradane" przez system operacyjny do czasowej obsługi operacji odczytu lub zapisu. Przyjrzyj się także, co się dzieje na ekranie podczas wciskania dowolnych klawiszy.

Klaudiusz Dybowski

```
46   100 REM *** COMMODORE 64 - ZERO PAGE ***
38   105 :
78   110 FOR J=49152 TO 49191:READ Q:POKE J,Q:NEXT
D4   115 SYS 49152:NEW
78   120 :
15   125 DATA 120,173,020,003,141,038,192,173,021,003
83   130 DATA 141,039,192,169,192,141,021,003,169,025
64   135 DATA 141,020,003,088,096,162,000,181,000,157
44   140 DATA 000,004,232,224,255,208,246,076,000,000
```

```
67   100 REM *** COMMODORE 128 - ZERO PAGE ***
38   105 :
4F   110 FOR J=3072 TO 3111:READ Q$:POKE J,DEC(Q$):NEXT
AC   115 SYS 3072:NEW
78   120 :
7A   125 DATA 78,AD,14,03,8D,26,0C,AD,15,03,8D,27,0C,A9
D5   130 DATA 0C,8D,15,03,A9,19,8D,14,03,58,60,A2,00,B5
55   135 DATA 00,9D,00,04,E8,E0,FF,D0,F6,4C,00,00
```

```
16   100 REM *** COMMODORE 16/116/+4 - ZERO PAGE ***
38   105 :
02   110 FOR J=818 TO 857:READ Q$:POKE J,DEC(Q$):NEXT
5F   115 SYS 818:NEW
78   120 :
6E   125 DATA 78,AD,14,03,8D,58,03,AD,15,03,8D,59,03,A9
3D   130 DATA 03,8D,15,03,A9,4B,8D,14,03,58,60,A2,00,B5
64   135 DATA 00,9D,00,0C,E8,E0,FF,D0,F6,4C,00,00
```

HELP SCREEN



Wszystkie programy profesjonalne, takie jak dBASE IV, LOTUS 1-2-3 czy CHIWRITER, wspomagają początkującego użytkownika tzw. „helpem" zawartym w samym programie. Wystarczy jedynie wcisnąć klawisz F1 czy ALT H i automatycznie na ekranie pojawia się tekst pomocniczy — np. lista poleceń programu.

Niektóre programy dla Commodore 128 i 64 (np. DATA MANAGER, SWIFTCALC czy MULTIPLAN 64) mają również taką opcję. Podany poniżej programik umożliwi Ci tworzenie Twoich własnych „ekranów" pomocniczych w dowolnym programie.

W rzeczywistości procedura tu opisana zamienia wymiennie dwa obszary pamięci — $0400 do $0800 (normalna pamięć ekranowa) oraz obszar od adresu $C100 do $C500 (49408—50432). Tekst pomocniczy powinien być umieszczony na ekranie 2. Program zawiera króciutką procedurę demonstracyjną (linie 175 — 235).

Tekst pomocniczy można wpisać dwojako: albo za pomocą PRINT albo np. poprzez normalne wpisanie tekstu i zapisanie całości za pomocą monitora języka maszynowego. Trzecia możliwość to zapisanie tekstu w liniach z instrukcjami DATA i wykonanie serii POKE wpisujących tekst bezpośrednio w podany obszar pamięci.

Za drugi obszar odpowiadają parami powtarzające się liczby 193—194—195—196. Obszar ten można zmienić na inny poprzez zamianę tych liczb. W podanej konfiguracji program może kolidować z popularnym programem TURBO TAPE 64.

Leonard Kasprzak.

```
EA   100 REM ***  HELP SCREEN 64  ***
38   105 :
E5   110 REM *** LEONARD KASPRZAK ***
F8   115 :
78   120 :
C8   125 B=49152:C$=CHR$(147)
93   130 READ A:POKE B,A:B=B+1:C=C+A
41   135 IF B<49217 THEN 130
25   140 IF C<> 7374 THEN PRINT "SPRAWDZ DANE.
         ":END
D7   145 DATA 162,000,189,000,004,188,000,193,
         157,000,193
1A   150 DATA 152,157,000,004,189,000,005,188,
         000,194,157
C8   155 DATA 000,194,152,157,000,005,189,000,
         006,188,000
56   160 DATA 195,157,000,195,152,157,000,006,
         202,208,214
CB   165 DATA 162,231,189,000,007,188,000,196,
         157,000,196
8E   170 DATA 152,157,000,007,202,224,255,208,
         238,096
0A   175 PRINTC$"TO JEST EKRAN GLOWNY ($0400-0
         800)"
7B   180 GOSUB 220
68   185 PRINTC$"TO JEST EKRAN POMOCNICZY ($C1
         00-C500)"
3A   190 GOSUB 220
12   195 PRINT:PRINT
FD   200 PRINT "WCISNIECIE F1 ZMIENIA EKRANY."
80   205 GET B$:IF B$="" GOTO 205
D3   210 IF B$=CHR$(133) THEN SYS 49152
A2   215 GOTO 205
A6   220 PRINT:PRINT "WCISNIJ DOWOLNY KLAWISZ"
B3   225 GET A$:IFA$=""GOTO225
E3   230 SYS 49152
25   235 RETURN
```

JUNIOR
I INNI

Problem komputeryzacji polskich szkół leży na sercu wielu osobom, lecz niestety nie tym, które powinny być najbardziej zainteresowane.

Z początkiem 1985 roku pojawił się w Polsce mikrokomputer ZX Spectrum. Wiązano z nim wielkie nadzieje, pojawiły się projekty wykorzystania Spectrumów w niemal każdej gałęzi życia. Jeden z projektów dotyczył szkół.

Jak zwykle zresztą, wszystko było w fazie domysłów, przygotowań i planów. Tylko bardziej przedsiębiorczy nauczyciele wydostali 100 tys. z funduszu pomocy naukowych zakupując Spectrum, magnetofon i parę taśm, czasem i telewizor.

Jak rzadkie były to działania niech powie fakt, że w samej Warszawie postąpiły tak początkowo dwie szkoły (licea ogólnokształcące). Potem odważnych przybywało, lecz rząd wielkości podniósł się znikomo.

Pojawił się problem inny: właściwego wykorzystania sprzętu. Najczęściej opiekun komputera (komputerów) dostawał wraz z nim instrukcję, z której uczył się i prowadził potem kółko „informatyczne". Bardzo rzadko trafiał się opiekun zaznajomiony trochę z Basicem, Odrą, Fortranem (na kartach perforowanych). A uczniowie łaknęli wiadomości. Najwięksi entuzjaści spędzali przy komputerze całe godziny, zarówno rano, przed lekcjami, jak i po nich. Uczyli się, doskonalili, pisali programy i nic z tego nie wynikało.

Komputer stał się bowiem „bogiem", a powinien być narzędziem. Na razie dostęp do niego mieli wybrańcy (czasem stał nawet w szafie pancernej lub w domu nauczyciela). Przebąkiwało się coś na temat programów edukacyjnych, testów i zastosowaniu komputera w nauczaniu, lecz sprowadzało się to najczęściej do programów „funkcja kwadratowa", „test z tabliczki mnożenia" lub „pięć dat historycznych".

Gdy pojawiły się gry, programowanie poszło w kąt. Pracownie komputerowe stały się salonami gier i trudno było temu delikatnie przeszkodzić.

Nieliczni jednak myśleli poważnie o zastosowaniu „Trumien" w edukacji. Montowali minisieci, telewizory w ścianach, pisali kompleksowe programy. Rzadko dawało to efekt, bo nie każdy zdaje sobie sprawę, jak doniosłą rzeczą jest rozumieć sens i cel działania, zanim się do niego przystąpi.

Okres stagnacji trwał. I co by nie robiono w szkołach z Commodorami, Amstradami i innym sprzętem, nie dawało to rezultatu.

Ogromnym wysiłkiem woli i innych ważnych czynników zaprojektowany został komputer ELWRO 800 JUNIOR, który miał uzdrowić informatycznie szkołę i nareszcie dać uczniom szansę poczucia się obywatelami Europy. Projekty, prototypy, plany, oferty — to wszystko dawało nam złudzenie, że coś się zmieni.

Zakłady Elektroniczne „Elwro" podjęły się wyprodukowania 5 tys. sztuk Juniorów do września 1987 roku. Na tej podstawie otrzymały zamówienia rządowe i rozpoczęły jego realizację.

Z przyczyn, których nie warto tu ani teraz rozstrząsać Juniory nie zalały szkół ani w 1987, ani 1988 roku. Formalnie wyekspediowana została ilość 5000 sztuk, nie zaspokoiło to pragnień i nadziei.

Tak więc przeciągnął się kolejny, tym razem poważny i nawet wiarygodny projekt komputeryzacji szkół. A uczniowie i nauczyciele czekali. Szczęśliwi, klepiąc w gumki „Spectrumny" marzyli o IBM-ach, nieszczęśliwi mieli nadzieję na wymarzone „Spectrum".

Tymczasem w Polsce powoli pojawiały się IBM-y. Bardzo nieliczne szkoły wysupłały milion lub dwa i zdołały zakupić ten symbol komputeryzacji, elektroniki, nowoczesności itp. — IBM PC XT, zwykle w najuboższej konfiguracji.

Nieliczni sięgali więc po wielki świat, a reszta — wciąż czekała.

I są tacy, którzy doczekali się. Otrzymali bardziej lub mniej kompletny „komplet" w postaci komputera nauczycielskiego, komputerów uczniowskich, stacji dysków, czase i drukarki. Mimo to, iż system sieci w szkole został uznany za zły już dawno i przez mądrzejszych od nas, to dałoby się z nim wytrzymać, ale...

No właśnie. W tym przypadku przeszkodą jest technika. Junior wielkości IBM-a „baby", drukarka wielkości kombajnu, plątanina kabli (dobrze, gdy są wszystkie) i: TO NIE DZIAŁA!

Kiepskie części, niedokładny montaż, ogólna bylejakość powoduje, że Juniory padają jak muchy. Ciągłe wizyty serwisu są nieodzowne przy instalacji niemal każdej pracowni „Juniorowej".

Jeśli już wszystko się uda, komputery działają a stacja warczy, pojawia się problem kolejny: obsługi tego dziwadła.

Komendy, klawisze specjalne, kombinacje zostały tak dokładnie wymieszane i skomplikowane, że odszukanie czegokolwiek bez pomocy instrukcji jest niemożliwe. Nie obejdzie się bez wertowania książeczki, prób i błędów. Ale to jest do opanowania; przed nami problem w istocie największy — oprogramowanie.

Wiadomo nie od dziś, że komputer jest wart tyle, co programy do niego. Jeśli mamy już pracownię, komputery, cały sprzęt działa, to co mamy zrobić? Pisać od nowa „funkcję kwadratową"? Dostarczane jako wyposażenie dyskietki z Basicem, Turbo Pascalem, Wordstarem to za mało. Równocześnie z rodzącym się w bólach Juniorem winno się było myśleć, co właściwie mamy zamiar z nim zrobić. Dobre oprogramowanie edukacyjne nie powstanie jeszcze długo, do tego trzeba wysiłków wielu ludzi i to od strony koncepcji, a nie tylko „siadł i napisał". Skoro mamy już sieci (pomijam ich sensowność i bezustanne kłopoty w użytkowaniu), komputery zgodne opcjonalnie z ZX Spectrum, lecz wykorzystywane, jak przystało na poważne systemy, pomyślmy, co zrobić, by przydatne były uczniom, by im coś dawały. Oni na TO czekają!

Marcin Przasnyski

Otrzymujemy listy od nauczycieli „uszczęśliwionych" sieciami „Juniorów". Proszą nas oni o radę, pomoc, kontakty. Niestety, jesteśmy prawie bezradni. To, co możemy zrobić w tej chwili, to zamieszczanie adresów szkół wyposażonych w „Juniory". Prosimy więc o listy.

Oto adres jednego z klubów:
Zespół Szkół Zawodowych WWK, 21-110 Ostrów Lubelski, ul. Armii Ludowej 3
Wyposażenie: sieć 9 komputerów, stacja 2 x 720 KB, drukarka D-100M.

POGRUBIONE LITERY

Jednym z mankamentów generatora znaków ZX Spectrum jest grubość liter, przeszkadzająca w wyrazistym wyświetlaniu danych na ekranie.

Poniższy 69-bajtowy programik powstaje właśnie, aby umożliwić pisanie pogrubionymi i standardowymi literami jednocześnie.

Nowy znak powstaje przez nałożenie na siebie jego zwykłej postaci i tej przesuniętej o jeden punkt w lewo (rozkazy maszynowe SRL i OR). Procedura ta jest relokowalna i można ją umieścić pod dowolnym adresem. Sposób wywołania: **LET zm=FN g („TEKST")**. TEKST może zawierać znaki o kodach od 32 do 127 włącznie, gdy wystąpią inne znaki, automatycznie wypisywanie tekstu zostanie przerwane. Program buduje pogrubioną literę jako **GRAPHICS „u"**, więc we własnym programie nie można definiować tego symbolu (po wykonaniu instrukcji pisania TEKST-u, **GRAPHICS „u"** zostanie wymazane).

Andrzej Wawrzyńczyk

```
10 INPUT "Adres startowy = ";adres: REM BE
20 LET suma=0: REM 48
30 FOR n=adres TO adres+68: REM A5
40 READ bajt: POKE n,bajt: REM 27
50 LET suma=suma+bajt: REM AC
60 NEXT n: IF suma<>6932 THEN STOP : REM 28
100 DEF FN g(a$)=USR adres: REM CA
110 PRINT AT 15,7;: LET zm=FN g("Jak sie wam podoba?"): REM 1E
120 PRINT AT 17,7;"chyba nie najgorzej": REM 11
200 DATA 42,11,92,35,35,35,35,126,95,35,126,87,35,70,235,94: REM 7C
210 DATA 123,254,31,216,254,127,208,22,0,197,229,33,160,0: REM 7A
220 DATA 237,75,123,92,9,235,41,41,41,237,75,54,92,9,6,8,126: REM 11
230 DATA 203,63,182,18,19,35,16,247,62,2,205,1,22,62,164,215: REM FC
240 DATA 225,35,193,16,203,201: REM D9
```

PCHŁA DRUGA

Ta pchła też brzęczy, ale przez generator AY, który już pewnie macie.

Wykorzystuje przy tym dwa porty, o numerach 65533 i 49149. Pełna definicja wszystkich możliwości — trzech kanałów, perkusji itp. nie zmieści się w tej pchle, poeksperymentujcie więc na razie w jednym kanale.

Po wciśnięciu spacji usłyszycie zmieniający się dźwięk, który zaniknie. Oprócz tego klawisze 1—9 pozwalają wywoływać krótkie nutki. I to wszystko, bo to tylko pchła.

Gen Martinez

```
10 LET a=65533: LET b=49149
20 OUT a,7: OUT b,255
30 OUT a,7: OUT b,61: OUT a,2: OUT b,0:
   OUT a,9: OUT b,15
40 LET a$=INKEY$
50 IF a$=" " THEN FOR f=120 TO 250:
   OUT a,2: OUT b,f: NEXT f: GO TO 20
60 IF a$>="0" AND a$<="9" THEN OUT a,2:
   OUT b,200+3*VAL a$
70 FOR f=1 TO 50: NEXT f
80 GO TO 20
```

LISTER

Prog amem tym chciałbym nawiązać do prze[illegible] w Bajtku [illegible] „Od środka” pp. Tomasza Surmacza i Roberta Dudnika. Czytelniku, czy nie spotkałeś się z podobną sytuacją: załadowałeś wreszcie loader np. jakiejś gry do pamięci tak, by nie wystartował, wykonujesz LIST i... no właśnie. Najczęściej zobaczysz „sieczkę”, listing wprost usiany jest różnego rodzaju znakami kontrolnymi, a co gorsze zawierający liczby z wartością zmiennoprzecinkową różniącą się od wartości „widzianej”. Nie jest się więc pewnym niczego.

Dla wielu jedynym wyjściem jest załadowanie programu pod COPY COPY, by tam usunąć znaki kontrolne oraz poprawić wartości liczb. Jest to jednak proces uciążliwy i czasochłonny. Lepszym rozwiązaniem byłoby wywołanie napisanej w kodzie maszynowym procedury, któraby listowała program w żądany sposób. Tym właśnie jest LISTER. Wykonuje on całą „brudną robotę” za nas, tzn. omija wszystkie kody z zakresu 0—31 i obok wszystkich liczb (z wyjątkiem traktowanych jako wyrażenie tekstowe, gdyż tu, to co się widzi, to „jest”) wypisuje w negatywie prawdziwą wartość. Pozwoliłem sobie połączyć swój program z LOAD/MERGE z Bajtka 2/88, gdyż jest to najkrótsza i najbardziej elegencka z możliwych procedura ładująca BASIC.

Aby wykorzystać mini-pakiecik **LOAD&LIST**, należy wpisać program w BASIC-u lub asemblerze (tu należy pamiętać o nagraniu programu na taśmę zleceniem **SAVE CHR$ 239** + „ & ” + **CHR$ 240 CODE ADRES, 101).**
By załadować loader danego programu, wykonujemy **RANDOMIZE USR adres początkowy,** by też go wylistować **RANDOMIZE USR adres początkowy** + **26**. Program jest w pełni relokowalny. Jeżeli chcemy wydrukować listing, to przed uruchomieniem należy wpisać **POKE adres początkowy** + **27,3**. (standardowo jest 2). Miłego LISTowania!

Jacek Marczewski

```
*HISOFT GENS3M2 ASSEMBLER*
        ZX SPECTRUM

Copyright (C) HISOFT 1983,4
All rights reserved

Pass 1 errors: 00

           10  *C-
           20  *D+
           30  ;   BASIC LISTER
           40  ;  (C)JAMSOFT'89
60000      50          ORG   60000
60000      60  LOADER  LD    BC,34
60003      70          RST   48
60004      80          PUSH  DE
60005      90          POP   IX
60007     100          LD    (IY+58),1
60011     110          LD    (IX+1),25
5
60015     120          CALL  1821
60018     130          LD    HL,(23618
)
60021     140          LD    (23621),H
L
60024     150          RST   8
60025     160          DEFB  255
60026     170  LISTER  LD    A,2
60028     180          CALL  #1601
60031     190          LD    HL,(23635
)
60034     200  NOWLIN  LD    A,(HL)
60035     210          CP    #40
60037     220          RET   NC
60038     230          CALL  #1A28
60041     240          INC   HL
60042     250          INC   HL
60043     260          INC   HL
60044     270  PETLA   LD    A,(HL)
60045     280          CALL  LICZBA
60048     290          CP    13
60050     300          JR    Z,KONLIN
60052     310          CP    32
60054     320          JR    C,DALEJ
60056     330          RST   16
60057     340          INC   HL
60058     350          JR    PETLA
60060     360  KONLIN  LD    A,13
60062     370          RST   #10
60063     380          INC   HL
60064     390          JR    NOWLIN
60066     400  DALEJ   INC   HL
60067     410          JR    PETLA
60069     420  LICZBA  CP    14
60071     430          RET   NZ
60072     440          INC   HL
60073     450          LD    A,32
60075     460          RST   #10
60076     470          LD    A,20
60078     480          RST   #10
60079     490          LD    A,1
60081     500          RST   #10
60082     510          PUSH  HL
60083     520          CALL  #33B4
60086     530          CALL  #2DE3
60089     540          POP   HL
60090     550          LD    A,20
60092     560          RST   #10
60093     570          LD    A,0
60095     580          RST   #10
60096     590          LD    A,(HL)
60097     600          INC   HL
60098     610          INC   HL
60099     620          INC   HL
60100     630          INC   HL
60101     640          RET

Pass 2 errors: 00

Table used:    102  from    198
```

```
 10 REM  PROGRAM LISTER
 20 REM  JACEK MARCZEWSKI
 30 REM
 40 CLEAR 59999: LET adres=6000
0: LET s=0
 50 FOR a=1 TO 4: FOR b=0 TO 24
 60 READ c: POKE adres+b,c
 70 LET s=s+c: NEXT b
 80 READ c: IF s<>c THEN PRINT
"Popraw dane w linii ";140+10*a:
STOP
 90 LET adres=adres+25: LET s=0
 100 NEXT a
 110 POKE adres,35: POKE adres+1
,201
 120 PRINT "LOAD - USR ";adres-1
00: PRINT "LIST - USR ";adres-74
 130 SAVE CHR$ 239+" & "+CHR$ 24
0CODE adres-100,102
 140 REM
 150 DATA 1,34,0,247,213,221,225
,253,54,58,1,221,54,1,255,205,29
,7,42,66,92,34,69,92,207,2681
 160 DATA 255,62,2,205,1,22,42,8
3,92,126,254,64,208,205,40,26,35
,35,35,126,205,165,234,254,13,27
69
 170 DATA 40,8,254,32,56,10,215,
35,24,240,62,13,215,35,24,224,35
,24,231,254,14,192,35,62,32,2366
 180 DATA 215,62,20,215,62,1,215
,229,205,180,51,205,227,45,225,6
2,20,215,62,0,215,126,35,35,35,2
962
```

NA CO CZEKAMY?

Popularność Spectrum nie słabnie. Co więcej, wzrasta, gdyż „Trumna” z całą baterią przystawek potrafi zmusić do opadnięcia szczękę niejednego Commodorowca.

Generatory dźwięku, stacje dysków, pamięć 80KB będą znajdować się niedługo w każdym egzemplarzu Spectrum, gdyż ich właściciele dbają, by nie wypaść z obiegu. Ale mimo tak znacznych możliwości Spectrum używane jest głównie do gier.

Dobrze poinformowani wiedzą, że 10 000 to tylko przybliżona liczba gier, będących w obiegu. Skąd ich tyle? Po prostu są wciąż produkowane. Gdy powstaje gra, to jednocześnie w kilku wersjach, najczęściej na Commodore, Amstrada, Atari ST, Amigę i oczywiście Spectrum.

W Wielkiej Brytanii, gdzie pisze się najwięcej gier na komputery ośmiobitowe, rynek odbiorców gier jest bardzo obszerny i stosunkowo największe zapotrzebowanie występuje ze strony posiadaczy Spectrum.

Tak więc co miesiąc ukazuje się na rynku około 20 nowych gier. Rodzimi handlarze sprowadzają je natychmiast przez swoje „wtyczki”, łamią i rozprowadzają dalej, głównie na giełdzie. Spora część gier nadchodzi jednak z opóźnieniem. Przyczyną jest cena: gra, gdy wychodzi po raz pierwszy, kosztuje ok. 15£, zaś po dwóch miesiącach już ok. 8£.

Próbując wyprzedzić niektóre gry, zamieszczamy tutaj listę ostatnich hitów z Anglii.

1. **Red Heat** — wieloetapowa strzelanina na podstawie filmu z Arnoldem Schwarzeneggerem. Ocena 74%.
2. **Forgotten Worlds** — również strzelanina, niezwykle szybka, barwna i urozmaicona. Ocena 90%.
3. **Chuck Yeager's Advanced Flight Trainer** — rewelacyjny symulator lotu, napisany przez firmę Electronics Arts w konsultacji z pilotem Chuckiem Yeagerem. Ciekawa grafika, 16 samolotów do wyboru, opcje nauki, figur akrobatycznych itp. Ocena 80%.
4. **Crazy Cars Europa** — kontynuacja gry Crazy Cars. Tym razem do wyboru kilkanaście tras przez Europę. Ocena 75%.
5. **Jaws** — komnatowo-przygodowa gra oparta na pomyśle z filmu „Szczęki”. Bardzo ciekawe efekty graficzne. Ocena 69%.
6. **Gemini Wing** — jeszcze jedna bezsensowna strzelanina, przeniesienia z automatów. Ocena 56%.
7. **License to Kill** — misja szpiega gdzieś w dalekim kraju. Do zdobycia dokumenty i spora gotówka. Ocena 70%.

Firma Ocean zapowiada grę Nightbreed, Domark gry Xybots i Sanchez, lecz na razie nie wiadomo o nich nic.

Gen Martinez

POPRAWKA DO GENERATORA DŹWIĘKU

W numerze 6/89 zamieściliśmy schemat ideowy generatora dźwięku AY 3-8910 do ZX Spectrum. W schemacie nie została opisana jedna nóżka chipa AY, położona w jego lewym górnym rogu. Powinna mieć ona numer 22. Za niedopatrzenie serdecznie przepraszamy.

Redakcja

KNIFE PLUS

EDYTOR DYSKOWY

The Norton Utilities Advanced Edition jest najbardziej znanym edytorem dyskowym komputerów serii IBM PC.

Programy tego typu umożliwiają przeglądanie i modyfikację zawartości dowolnego dysku. W wielu sytuacjach awaryjnych pełnią one rolę ostatniej deski ratunku, pozwalając odtworzyć niedomknięty lub usunięty zbiór. Osoby bardziej zaawansowane, ale często mniej moralne, korzystają z edytorów dyskowych do „spolszczania" zagranicznych programów użytkowych lub gier.

IBM PC nie jest wyjątkiem, praktycznie na każdym komputerze z dyskowym systemem operacyjnym dostępny jest program umożliwiający edycję dyskietki. W przypadku CP/M Plus dla 8-bitowych Amstradów firma HiSoft oferuje produkt pod nazwą Knife Plus, w wersji zarówno na PCW, jak i na CPC.

Na rysunku 1 przedstawiono podstawowy ekran edycji. W drugim wierszu znajdują się informacje o typie napędu (Drive: B), numery ścieżki (Track), sektora (Sector), bloku (Block) i nazwa pliku (file). Poniżej wyświetlona jest zawartość pierwszej połowy sektora w formie szesnastkowej (środkowa część) i kodów ASCII odpowiadających kolejnym bajtom. Klawisze strzałek pozwalają na ruch kursora, a tabulator (TAB) przełącza tryby edycji (szesnastkowy i ASCII).

Znaczenie pozostałych klawiszy opisane jest na rysunku 2, przedstawiającym ekran „pomocy" programu Knife+, dostępny po naciśnięciu klawiszy ALT H. Znajomość kilku angielskich słów kluczowych pozwala zorientować się w możliwościach oferowanych przez program, bez korzystania z instrukcji. Oprócz wyświetlania i modyfikacji dowolnych sektorów, bloków i fragmentów pliku, bardzo ciekawą opcją jest wyszukiwanie ciągów znaków (lub bajtów) w dowolnym miejscu dysku. Utworzenie niedomkniętego, w skutek awarii komputera lub programu, pliku jest możliwe dzięki poleceniom uzyskanym po sekwencji klawiszy EXTRA B, EXTRA R i EXTRA C.

Uzupełnieniem zbioru KNIFE+ COM jest kilka dodatkowych plików znajdujących się na firmowej dyskietce HiSoft'a. Krótki program SD.COM służy do wyświetlenia katalogu dyskietki z podaniem dokładnych informacji o plikach, a zbiór UNERA.COM pozwala odzyskać skasowany omyłkowo plik. W wielu systemach dyskowych usunięcie pliku polega na zmianie jednego bajtu w skorowidzu (ang directory) dyskietki, bez fizycznego kasowania zapisanych sektorów. Z tego powodu działanie programu UNERA.COM nie jest szczególnie skomplikowane.

Czasami przy pracy, nawet z firmowymi dyskietkami (szczególnie Maxell'a), zdarzają się problemy z odczytaniem jakiegoś sektora, co nie pozwala na skopiowanie lub uruchomienie programu. Jeśli po wielu próbach uda się wczytać taki wątpliwy sektor do pamięci komputera, przy pomocy programu Knife+, to jego ponowne zapisanie może „naprawić" dyskietkę. Po odtworzeniu zbioru i zapisaniu go na inną dyskietkę, należy од nowa sformatować tę wadliwą albo pozbyć się jej.

Reasumując, uważam, że program Knife Plus, mimo pewnych braków (nie ma mapy alokacji sektorów i poszczególnych plików na dysku) jest bardzo użytecznym narzędziem i powinien stanowić awaryjne wyposażenie każdego użytkownika Amstrada.

Na zakończenie krótki słownik podstawowych terminów, ułatwiający zorientowanie się w możliwościach programu KNIFE+, przedstawionych na rys. 2.

goto — idź do, *previous* — poprzedni *next* — następny, *select* — wybierz, *new* — nowy, *write* — zapisz *paste* — zapamiętaj, aby zapisać w innym miejscu, *search* — szukaj, *pattern* — wzorzec, *show* — pokaż, *half* — połowa, *flip between* — przełącz pomiędzy *quit* — wyjdź, *filename* — nazwa zbioru, *build* — zbuduj (utwórz) *close* — zamknij.

Jonasz Mayer

Opisywany program został nam udostępniony przez pana Jacka Cieślikowskiego. Dziękujemy.

```
CP/M Plus Knife Plus Disc Sector Editor V3.03 (C) HiSoft 1987. PCW range.
 Drive: B     Track: 4B    Sector: 07   Block: 00A8   File: KNIFE+  .COM

0000  18 3A B3 17 20 4B 6E 69 66 65 20 50 6C 75 73 20   .:3. Knife Plus
0010  28 43 29 20 48 69 53 6F 66 74 20 32 30 20 53 65   (C) HiSoft 20 Se
0020  70 20 38 37 20 50 4C 45 41 53 45 20 42 55 59 20   p 87 PLEASE BUY
0030  44 4F 4E 27 54 20 53 54 45 41 4C 20 ED 7B 06 00   DON'T STEAL m{..
0040  CD A2 11 3E 01 32 DF 18 3D 32 DD 18 32 DE 18 32   M".>.2_.=2].2↑.2
0050  E4 18 32 52 19 32 D5 18 32 D1 18 3D 32 F2 18 0E   d.2R.2U.2Q.=2r..
0060  19 CD 05 00 32 D7 18 3A E3 18 FE 22 3E FF 28 08   .M..2W.:c.~">.(.
0070  5F 0E 2D CD 05 00 18 03 CD 80 BE C5 D5 E5 CD 6C   _.-M.....M.>EUeMl
0080  11 E1 D1 C1 11 48 12 CD 65 05 11 93 12 3A E3 18   .aQA.H.Me....:c.
0090  FE 22 28 03 11 97 12 CD 65 05 11 4E 12 CD 65 05   ~"(....Me..N.Me.
00A0  AF 32 CF 18 CD 0E 02 20 D2 3A E3 18 FE 22 28 09   /2O.M.. R:c.~"(.
00B0  1E FE 0E 2D CD 05 00 18 04 AF CD 80 BE C5 D5 E5   .~.-M..../M.>EUe
00C0  CD 6C 11 E1 D1 C1 3A D7 18 CD 48 0E 11 D5 15 0E   Ml.aQA:W.MH..U..
00D0  0F CD 05 00 3C CA 00 00 21 ED B0 22 FE 00 21 F7   .M..<J..!m0"~.!w
00E0  2B EB D5 0E 10 CD 05 00 11 D5 15 0E 14 CD 05 00   +kU..M...U...M..
00F0  D1 21 80 00 19 B7 28 E9 11 F7 2B ED 52 44 4D 21   Q!...7(i.w+mRDM!
[illegible]
```

Rys. 1 Ekran edycji pliku

```
[illegible]

Use cursor keys to move around edit window
and the [TAB] key to move between hex and ASCII.

[illegible]          goto previous sector
[illegible]          goto next sector
[BOXED]-MINUS        goto previous track
[BOXED]-PLUS         goto next track
[ALT]-T              select new track
[ALT]-S              select new sector
[ALT]-W              write this sector - CARE!
[ALT]-P              paste this sector
[EXTRA]-[illegible]  search for ASCII pattern
[ALT]-C              search for byte pattern
[ALT]-N              search for next pattern
[EXTRA]-A            show other half of sector
[TAB]                flip between hex and ASCII edit
[STOP]               quit edit
[ALT]-B              select new block
>                    goto next block
<                    goto previous block
[illegible]          goto filename
[EXTRA]-B            build this block into build file
[EXTRA]-R            build this sector into build file
[EXTRA]-C            close build file
[ALT]-H              show help

Press a key ...
```

Rys. 2 Ekran „pomocy" programu Knife+

PAKIET PROCEDUR FIRMY CP SOFTWARE

Na początku 1986 roku na rynku angielskim pojawił się Amstrad PCW 8256. Niewielka ilość oprogramowania dostępna w tym czasie, utrudniała jego wykorzystanie. Uboga literatura dotycząca sprzętu też nie ułatwiała życia programistom. Szczególnie kłopotliwy był brak instrukcji graficznych, sprzedawanego razem z komputerem, interpretera języka BASIC. Rozwiązaniem tych problemów jest oferta brytyjskiej firmy CP Software, proponującej pakiet 52 procedur w kodzie maszynowym, dostępnych w BASIC'u.

Dyskietka firmowa

Na rysunku 1 przedstawiono katalog dyskietki firmowej, zawierający 11 plików. Tekst źródłowy pakietu, przygotowany w języku asemblera procesora Z80 znajduje się w zbiorze SCODE.GEN. Wersję skompilowaną, nadającą się do uruchomienia, zawiera zbiór SCODE.COM. Pliki z roszerzeniem BAS, są przykładowymi programami w BASIC'u, demonstrującymi wykorzystanie pakietu. Uruchomienie programu DEMO.COM pozwala na bezpośrednie przekonanie się o możliwościach oferowanego zestawu procedur.

Lista procedur

Listę procedur pakietu (TAB. 1) można podzielić na siedem głównych grup. W grupie pierwszej znajdują się procedury pozwalające na uzyskanie efektów dźwiękowych przy pomocy zainstalowanego w komputerze brzęczyka. Druga grupa zawiera 13, stosunkowo prostych, procedur obsługi trybu tekstowego ekranu. Każdą z nich można bezproblemowo zastąpić w BASIC'u instrukcją PRINT z odpowiednimi parametrami. Kolejna grupa zawiera 4 procedury graficzne rysujące na ekranie punkty i proste (PLOT, DRAW, UNPLOT, UNDRAW). Obsługę ekranu zapewnia zestaw 11 procedur, z których na uwagę zasługuje, przydatna w grach procedura wyświetlania duszka (ang. sprite). Inna interesująca procedura wypełniania wnętrza figury nie działa dobrze. Piąta grupa

procedur służy do drukowania na ekranie i drukarce liter o różnych szerokościach i wysokościach. Możliwe jest również zdefiniowanie własnego generatora znaków. Procedury systemowe zawarte w szóstej grupie obsługują między innymi własny jednobajtowy generator liczb losowych, czytają joystick i realizują dostęp do zegara systemowego. Ostatnia grupa zawiera 4 procedury dotyczące stacji dysków i plików na dyskietce.

Współpraca z BASIC'em

Przed wywołaniem interpretera języka BASIC należy uruchomić program SCODE.COM, który instaluje pakiet w pamięci komputera w przestrzeni adresowej: $COOO_H$ — $E73E_H$. Pierwszą instrukcją, konieczną do wykonania w BASIC'u jest MEMORY &HBFFF, która rezerwuje pamięć dla pakietu, ale jednocześnie zmniejsza obszar dostępny na własny program. Korzystanie z procedur jest stosunkowo proste i polega na zastosowaniu kombinacji instrukcji POKE i CALL. Przykładowo wyświetlenie na ekranie punktu o współrzędnych 50,100 wymaga następującej sekwencji poleceń:

A = 49190 : POKE 49220,50 : POKE 49221,0 : POKE 49222,100 : POKE 49223,0 : CALL A

Współpraca z Turbo Pascalem

W dokumentacji pakietu opisano wykorzystanie pakietu tylko z poziomu BASIC'a. Jednak przy zachowaniu pewnej ostrożności można również odwoływać się do niego przy pracy w Turbo Pascalu. W zasadzie bezpieczne są tylko programy kompilowane na dysk z adresem końcowym (ang. END ADDRESS) mniejszym od $COOO_H$. Utrudnia to uruchamianie dłuższych programów. Krótkie procedury nie deklarujące dużej pamięci na zmienne, mogą być kompilowane do pamięci. Dzięki instrukcji INLINE, tablicy MEM lub zmiennym z atrybutem ABSOLUTE, korzystanie z pakietu w Pascalu. jest znacznie prostsze niż w BASIC'u. Podano dwa krótkie programy demonstrujące zastosowania pakietu przy pracy z Turbo Pascalem. Listing 1 zawiera program kreślący trójkąt i wypełniający jego wnętrze. W programie MUSIC (listing 2) zdefiniowano procedurę NOTE, która pozwala na wygranie określonego dźwięku na brzęczyku komputera.

Podsumowanie

Prezentowany pakiet firmy CP Software zawiera kilkadziesiąt niezwykle użytecznych procedur, których zastosowanie zarówno w BASIC'u, jak i Turbo Pascalu, pozwala na znaczne uatrakcyjnienie własnych programów. Dobra dokumentacja i bogato opisana wersja źródłowa procedur, znajdująca się w zbiorze SCODE.COM, stanowi doskonałą kopalnię wiedzy o posiadanym komputerze i może być przedmiotem samodzielnych studiów dla osób bardziej zaawansowanych. Moją szczególną uwagę zwróciła duża szybkość procedur graficznych i możliwości dźwiękowe pakietu.

Jonasz Mayer

Opisywany pakiet został nam udostępniony przez pana Jacka Cieślikowskiego. Dziękujemy.

TAB. 1.

Lista procedur:

(numer procedury i wykonywana operacja)

I. DŹWIĘK

1. Włącz brzęczyk
2. Wyłącz brzęczyk
3. Brzęczyk (krótki sygnał)
48. Wygraj dźwięk o zadanej częstotliwości i długości

II. TRYB TEKSTOWY EKRANU

4. Kursor w pozycji home (lewy górny róg ekranu)
5. Wyłącz kursor
6. Włącz kursor
7. Zapamiętaj pozycję kursora
8. Odtwórz zapamiętaną pozycję kursora
9. Kursor o linię do góry
10. Kursor o linię na dół
11. Kursor w lewo
12. Kursor w prawo
13. Ustaw pozycję kursora
14. Czyść ekran
15. Cofnij kursor i usuń znak
16. Wykonaj powrót karetki

III. PROSTA GRAFIKA

17. Narysuj linię prostą (lub kilka linii)
18. Usuń linię prostą (lub kilka linii)
19. Wyświetl punkt (PLOT)
20. Zgaś punkt (UNPLOT)

IV. OBSŁUGA EKRANU

29. Inwersja ekranu
30. Odtworzenie pierwotnej postaci ekranu (jeśli była inwersja)
31. Zgaś ekran
32. Zapal ekran z powrotem
41. Wypełnij linię ekranu (Fill a screen line)
42. Wypełnij wnętrze figury
46. Inicjalizacja duszka (sprite)
47. Wyświetl duszka
50. Przesuń (scroll) cały ekran w górę lub w dół
51. Załaduj obraz z dysku
52. Zapisz obraz na dysk

V. TEKSTY W TRYBIE GRAFICZNYM

36. Wydrukuj łańcuch znaków (trzy wielkości)
37. Wydrukuj n odstępów (spacji)
38. Zdefiniuj własny znak w generatorze znaków
39. Przełącz między czcionką pochyłą a zwykłą
40. Przełącz między czcionką 2001 a zwykłą
49. Wydrukuj na drukarce tekst zawierający samodzielnie zdefiniowane znaki

VI. PROCEDURY SYSTEMOWE

21. Znajdź zegar systemowy
22. Usuń linię statusu (Drive is ...)
23. Odtwórz linię statusu
24. Inicjalizuj generator liczb losowych
25. Pobierz liczbę losową
26. Opóźnienie o n sekund
27. Czekaj na powrót plamki (frame flyback)
28. Czytaj joystick
45. Pobierz znak z klawiatury
35. Wyzerowanie systemu (RESET)

VII. OBSŁUGA DYSKÓW I PLIKÓW

33. Włącz napęd stacji dysków
34. Wyłącz napęd stacji dysków
43. Załaduj plik do pamięci
44. Zapisz fragment pamięci na dysku

```
Directory For Drive A:  User  0

    Name     Bytes  Recs   Attributes     Name     Bytes  Recs   Attributes
------------ ------ ------ ------------ ------------ ------ ------ ------------
DEMO     COM    1k      3 Dir RW      DEMO1    COM   11k     82 Dir RW
DEMO2    COM    4k     28 Dir RW      DEMO3    COM    3k     18 Dir RW
EXAMPLE1 BAS    1k      4 Dir RW      EXAMPLE2 BAS    1k      8 Dir RW
EXAMPLE3 BAS    1k      5 Dir RW      EXAMPLE4 BAS    1k      6 Dir RW
EXAMPLE5 BAS    1k      4 Dir RW      SCODE    COM   10k     80 Dir RW
SCODE    GEN   54k    426 Dir RW

Total Bytes     =     88k  Total Records =     664  Files Found =   11
Total 1k Blocks =     88   Used/Max Dir Entries For Drive A:   14/  64
```

Rys. 1. Katalog dyskietki firmowej.

```
Program Line_and_Fill;

procedure Xclrscr;
begin
   inline ($CD/$55/$C2);
end; { of Xclrscr }

procedure DrawLine (x1,y1,x2,y2 : integer);
begin
   Mem [$DB7A] := 1;        Mem [$DB7B] := 0;
   Mem [$DB7C] := 0;      ; Mem [$DB7D] := 0;
   Mem [$DB7E] := 0;      ; Mem [$DB7F] := 0;
   Mem [$DB80] := Lo(x1);   Mem [$DB81] := Hi(x1);
   Mem [$DB82] := Lo(y1);   Mem [$DB83] := Hi(y1);
   Mem [$DB84] := Lo(x2);   Mem [$DB85] := Hi(x2);
   Mem [$DB86] := Lo(y2);   Mem [$DB87] := Hi(y2);
   inline ($CD/$00/$C0);
end; { of DrawLine }

procedure FillArea (x,y : integer);
begin
   y := 255-y;
   Mem [$DB73] := Lo(y);
   Mem [$DB74] := Lo(x);   Mem [$DB75] := Hi(x);
   inline ($CD/$FE/$E3);
end;

var
   ch : char;

begin
  Xclrscr;
  DrawLine (100,100,150,100);
  DrawLine (100,100,150,150);
  DrawLine (150,150,150,100);

  repeat until keypressed; read(kbd,ch);

  fillarea (120,102);
end.
```

Listing 1. Plik DRAWLINE.PAS

```
Program Music;

procedure Note (pitch, duration : integer);
begin
   Mem [$E46A] := Lo(pitch);      Mem [$E46B] := Hi(pitch);
   Mem [$E468] := Lo(duration);   Mem [$E469] := Hi(duration);
   inline ($CD/$44/$E4);
end; { of DrawLine }


var i,j : integer;

begin
  for i:= 1 to 10
  do for j := 1 to 1

     do begin
            note  (i*500,j);
            delay (100);
        end;
end.
```

Listing 2. Plik MUSIC.PAS

JĘZYK C DLA NAJMŁODSZYCH

czyli trzeci wieczór z czarnoksiężnikiem

Następnego wieczoru nasz znajomy usiadł wygodnie w fotelu. W chwili, gdy sięgał po czarnoksięską księgę zadzwonił telefon. Po rozmowie telefonicznej naszemu przyjacielowi było bardzo wstyd. Rozmówca przypomniał mu, że właśnie minęły imieniny ich wspólnego kolegi a on nawet się nie odezwał.

— Jak to się mogło stać — przecież ja mam niezawodny sposób? — pomyślał. Wyciągnął z kieszeni chustkę na której było dla pamięci zawiązanych kilka supełków. Niestety, nie pamiętał co te supełki miały przypominać.

— To nie jest dobry sposób — pomyślał. Mogę przecież zrobić sobie taki notatnik w pamięci komputera. Jest to jednak praca na kilka (może kilkanaście) wieczorów. Ponadto, zanim zaprojektuję strukturę takiego notatnika oraz sposób korzystania z niego, powinienem poznać więcej zaklęć i skonstruować kilka użytecznych algorytmów. Na początek przydałoby się, żebym potrafił określić długość wprowadzanego do komputera nazwiska lub ogólnie tekstu.

Sięgnął po księgę, otworzył ją i zaczął czytać:

„Znaki tekstu przechowywane są w pamięci komputera w kolejnych jej komórkach. W celu łatwego znalezienia końca tego tekstu, za ostatnim znakiem umieszczony jest zerowy znak '0' (analogicznie do 'n' jako nowej lini). Tekst zajmuje więc o jedną komórkę pamięci więcej niż ma znaków. Stałe tekstowe umieszczamy w cudzysłowach dla odróżnienia od stałej wartości, którą umieszczamy w apostrofach. "x" jest tekstem zajmującym dwie komórki pamięci — kod litery x i zerowy znak 0. 'x' jest stałą wartością równą kodowi litery x zajmującą jedną komórkę pamięci.

— Jest to bardzo istotna różnica, o której muszę pamiętać, gdy będę pisał programy przetwarzające teksty — pomyślał czarnoksiężnik.

Teraz zamyślił się przez chwilę.

— Algorytm jest prosty. Muszę sprawdzić, czy znaki są różne od zera, jednocześnie zwiększając numer znaku o jeden. W ten sposób sprawdzę kolejne znaki aż dotrę do znaku zerowego, który kończy tekst. Sprawdzanie muszę powtarzać w pętli. Jeden sposób zapętlenia już znam. Ciekawe, czy istnieją inne sposoby?

Otworzył księgę i kontynuował czytanie:

„Kolejnym sposobem zapętlenia programu jest wykonanie instrukcji w pętli „robgdy". Ma ona postać:

```
robgdy (warunek) instr;
```

W pętli tej „instr" wykonuje się tak długo, jak długo „warunek" jest prawdziwy. „Instr" może być pusta tzn. może jej nie być. W przypadku, gdy chcemy w pętli wykonać kilka instrukcji, wtedy „instr" jest blokiem kilku instrukcji ujętym w nawiasy klamrowe { }. Przypominam, że każda instrukcja musi kończyć się średnikiem;."

— Ten sposób będzie teraz bardziej odpowiedni — pomyślał. — Najpierw jednak muszę uzupełnić zbiór „zaklęć". —

Do utworzonego wcześniej zbioru ‹predef.h› dopisał

```
# define robgdy while
```

a następnie zaczął tworzyć tekst programu:

```
# include ‹stdio.h›
# include ‹predef.h›
majster
tekst t[100];
calkowite licznik = 0;
{czytaj („%s", t);
robgdy (t[licznik++]!='\0');
pisz ("\nTen tekst zajmuje %d pamięci", licznik);}
```

W programie tym występują dwie ciekawe konstrukcje. Deklaracja zmiennej licznik ma postać:

```
calkowite licznik =0;
```

co spowoduje nie tylko rezerwację pamięci dla tej zmiennej ale także nadanie jej wartości początkowej 0.

Drugą ciekawostką jest konstrukcja w pętli „robgdy". W pętli tej „instr" jest puste. Wszystkie czynności wykonują się przy okazji sprawdzania „warunku". Czynnością tą jest zwiększanie licznika „licznik" o jeden. Osiągnięto to dzięki zapisowi „licznik++", który powoduje po każdorazowym pobraniu wartości zmiennej „licznik" zwiększenie jej o jeden. Jak widać pozwala to pisać programy w bardzo zwięzłej formie. Może to jednak prowadzić do pogorszenia czytelności programu.

Umieszczenie plusów przed nazwą zmiennej "++licznik" spowoduje najpierw zwiększenie jej wartości o jeden, a następnie udostępnienie jej do dalszych obliczeń.

Konstrukcja ta dotyczy także minusów:

licznik-- udostępnia wartość zmiennej licznik do obliczeń, a następnie zmniejszy ją o jeden

--licznik zmniejsza wartość zmiennej licznik o jeden, a następnie udostępnia ją do dalszych obliczeń.

Mechanizmy te nazywamy inkrementacją

licznik++ postinkrementacja

++licznik preinkrementacja

oraz dekrementacją

licznik-- postdekrementacja

--licznik predekrementacja.

Powróćmy jednak do sprawy pętli. Dysponując pętlą „robgdy" możemy nie używać pętli „powtarzaj". łatwo zauważyć, że konstrukcja:

```
powtarzaj (start; warunek; zmiana)
instrukcja;
```

jest równoważna konstrukcji:

```
start;
robgdy (warunek)
{ instrukcja;
zmiana;}
```

Wybór użytej konstrukcji zależy od woli i upodobań twórcy programu. Dla porównania, program, który powstał podczas poprzedniego spotkania, przeliczający mile na kilometry, może mieć także taką postać:

TRANSPUTER DO XT



Na początku roku 1987, kiedy pojawił się nowy 32 bitowy mikroprocesor firmy Intel o symbolu 80386, można było przypuszczać, że rok ten w historii komputerów okaże się kolejnym okresem dominacji maszyn koncernu IBM, a szczególnie modelu 80 nowej serii PS/2.

Pojawienie się nowych transputerów T414 i T800 brytyjskiej firmy INMOS zmieniło ten obraz. Za 3000 funtów można było kupić kartę B004, która po włożeniu do starego modelu IBM PC/XT zwiększała jego wydajność obliczeniową 20—25 razy. Jest to kilkakrotnie większy przyrost niż ten, który zapewnia zestaw 80386 i 80387.

Wysoka cena tej karty, poważnie ograniczała popularność tego rozwiązania, do czasu pojawienia się konkurencyjnej w cenie karty Monoputera firmy MicroWay. Za dwa razy mniejsze pieniądze w połowie roku 1987 można było nabyć funkcjonalny odpowiednik karty firmy INMOS. Karta Monoputera zawiera 20 MHz wersję transputera T800 i 2MB pamięci RAM. Dołączone oprogramowanie, łącznie z kompilatorem języka Occam i programami pomocniczymi pozwala na wykorzystanie olbrzymich możliwości transputera na komputerze typu IBM PC.

Samo słowo transputer jest złożeniem dwóch słów angielskich *trans*istor i com*puter* i oddaje filozofię tego układu, który ma być wykorzystywany jako podstawowy element przy projektowanniu dużych współbieżnych systemów, tworzonych z wielu połączonych transputerów.

Układ T800 zawiera 32-bitowy procesor, koprocesor numeryczny operujący na słowach 64-bitowych, 4kB szybkiej statycznej pamięci RAM i cztery układy wejścia/wyjścia. Dzięki tym ostatnim, możliwe jest łączenie wielu transputerów w sieć o dowolnej konfiguracji, w której każdy procesor ma co najwyżej czterech sąsiadów. Tańsza wersja T414 nie posiada koprocesora numerycznego, ale nie różni się niczym innym i jest zamienna z T800 na płycie Monoputera.

Podstawowym językiem programowania dla transputerów jest jeden z pierwszych języków programowania współbieżnego Occam. Filozofia tego, nadzwyczaj zwięzłego w środkach języka (patrz brzytwa Ochama) opiera się na dwóch pojęciach — procesu i kanału (ang. process, channel). Procesor wykonuje współbieżnie lub sekwencyjnie procesy, które komunikują się między sobą przy pomocy kanałów.

Transputer mimo możliwości programowania w assemblerze jest w zasadzie sprzętową implementacją języka Occam. Wykorzystanie tego języka jest na razie najefektywniejszą metodą pracy na tym sprzęcie, ponieważ dostępne obecnie kompilatory innych języków wysokiego poziomu: C, Fortran, Pascal generują znacznie wolniejszy (do 10 razy) kod wynikowy. Jak zwykle rozwój oprogramowania nie jest tak szybki jak rozwój sprzętu. Musimy trochę jeszcze poczekać, jeśli zamierzamy korzystać z języków do których przywykliśmy.

Niemniej sam Occam nie jest specjalnie trudny i osoby znające Pascal bez większych problemów mogą się go nauczyć. Znajomość Pascala jest także ważna z innych powodów. Prototypowe oprogramowanie sprzedawane łącznie z kartą Monoputera nie zapewnia wygodnego korzystania z zasobów głównego komputera, jakim jest IBM PC. Dostęp do zbiorów dyskowych, klawiatury i ekranu prościej jest zrobić przy pomocy własnego programu pracującego pod kontrolą systemu operacyjnego MS-DOS. Para: Turbo-Pascal 4.0 na IBM i Occam-2 na transputerze tworzy obecnie najwygodniejsze i najtańsze środo-

```
# include <stdio.h>
# include <predef.h>
majster
calkowite mile;
rzeczywiste kilometry;
{ czytaj ("%!", mile);
robgdy (mile!=0)
  { kilometry = mile 1.852;
  pisz ("%d mile=%f kilometry", mile, kilometry);
    czytaj ("%d", mile);}
  }
```

Widać, że poprzednia forma jest bardziej czytelna i zwarta.

W programie tym można dokonać jeszcze jednej zmiany. Eliminując zmienną „kilometry" można zmniejszyć ilość zmiennych. Program będzie miał postać:

```
# include <stdio.h>
# include <predef.h>
  majster
    calkowite mile;
    { powtarzaj (czytaj("%",, mile); mile !=0; czytaj ("%d", mile))
        pisz ("%d mile =%f kilometry",mile,mile 1.852);}
```

Która z przedstawionych wersji jest najbardziej odpowiednia (zwarta i czytelna). Wszystko zależy od biegłości we władaniu tym językiem oraz od indywidualnych upodobań programisty. Oczywiście, szybciej wykonują się te programy, w których nie oblicza się wielokrotnej wartości tego samego wyrażenia. Obliczenia należy wykonać jeden raz i wynik zapamiętać jako wartość pewnej zmiennej. Jeśli jednak pewne wyrażenie liczone jest tylko jeden raz, wtedy szkoda tracić miejsce w pamięci na rezerwację dla takiej zmiennej. Wartość takiego wyrażenia może być argumentem funkcji. Widać to w przykładzie, w którym zrezygnowano ze zmiennej „kilometry". Zmienna ta była argumentem funkcji „pisz". Zamiast niej zostało wstawione wyrażenie obliczające kilometry. Wydaje się jednak, że po tej zmianie, program jest mniej czytelny, bowiem nazwa zmiennej sugerowała jej zastosowanie.

— Ciekaw jestem, czy są jeszcze jakieś sposoby zapętlenia programu — pomyślał czarnoksiężnik. W tym momencie zegar zaczął wybijać godzinę 22.

— Jutro o tym poczytam, teraz czas na spanie — pomyślał wkładając piżamę.

(mp)

wisko rozwojowe do pracy. Pascal zapewnia obsługę operacji we/wy i przekazuje dane do transputera, który po wykonaniu obliczeń odsyła wyniki z powrotem. Efektem tej współpracy jest 20—25 razy większa szybkość przetwarzania w stosunku do 10MHz wersji IBM PC/XT z koprocesorem 8087-1.

Na pojedyńczym transputerze można także uruchamiać współbieżne programy, które po przeniesieniu na układ wieloprocesorowy będą działać znacznie szybciej. Karta Quadputera oferowana również przez firmę Microway (obecnie ca 20 000 $) zawiera cztery połączone ze sobą transputery T800, z których każdy korzysta z 4MB pamięci RAM. Całkowita moc obliczeniowa układu dla programów w języku Occam odpowiada mniej więcej 1/6 mocy superkomputera Cray-2. Dzięki efektywnemu wykorzystaniu architektury typu RISC oraz idei współbieżności procesów należy uznać transputer za bardzo udaną konstrukcję lat dziewięćdziesiątych dostępną już teraz.

Jonasz Mayer

```
PROC INMOS.ENTRY.POINT ()
  -- KOMENTARZ
  -- przykładowy program w języku Occam-2
  -- współpracuje z programem w Pascalu na
  -- IBM PC, który przekazuje dwie dane i
  -- odbiera ich sumę.

-- DEKLARACJE KANALOW
CHAN OF ANY ToPC, FromPC:
PLACE ToPC AT 0:                    -- do PC

PLACE FromPC AT 4:                  --z PC
-- DEKLARACJE ZMIENNYCH
INT Operand1, Operand2, Result:

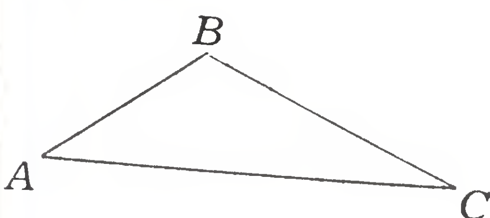
-- INSTRUKCJA SEKWENCYJNA
SEQ
    FromPC ? Operand1
    FromPC ? Operand2
    Result := Operand1 + Operand2
    ToPC :   ! Result
```

OKRĄG ZADANY PRZEZ 3 PUNKTY

TEORIA

Najłatwiej narysować jest okrąg, gdy dysponujemy współrzędnymi jego środka, x0, y0, i jego promieniem, r. Często jednak zdarza się, że mamy podane tylko 3 punkty, przez które przechodzi okrąg. W takim wypadku konieczne jest znalezienie parametrów okręgu: x0, y0, r.

Z geometrii na poziomie szkoły podstawowej wiadomo, że środek okręgu przechodzącego przez trzy punkty A, B, C (rys. 1) o współrzędnych x1,y1, x2,y2, x3,y3, znajduje się na przecięciu trzech symetralnych boków trójkąta wyznaczonego przez te trzy punkty.



Z punktu widzenia geometrii analitycznej, problem znalezienia środka okręgu w tym wypadku sprowadza się do rozwiązania układu 2 równań liniowych, z których pierwsze opisuje prostą, prostopadłą do jednego boku trójkąta i przechodzącą przez jego środek, a drugie taką samą prostą dla drugiego boku.

Rachunki są elementarne, ale dość długie i żmudne. Poniżej prezentujemy gotowe rozwiązanie:

$$s = \frac{(y3-y1)*(y3-y2)+(x3-x2)*(x3-x1)}{(x2-x1)*(y3-y2)-(x3-x2)*(y2-y1)}$$

$$x0 = 0.5 * ((x1+x2) - (y2-y1) * s)$$
$$y0 = 0.5 * ((y1+y2) + (x2-x1) * s)$$

Otrzymane wyniki, a szczególnie wzór na s, mają ciekawą interpretację. Mianownik ułamka piętrowego przedstawia pole trójkąta wyznaczonego przez trzy punkty A, B, C. Jeśli pole to jest równe zeru, to oznacza to współliniowość punktów i szukanie nieskończonego promienia nie ma sensu. Natomiast licznik opisywanego wyrażenia jest iloczynem skalarnym wektorów AC i BC. Jeśli iloczyn ten wynosi zero to oznacza, że kąt między tymi wektorami jest prosty i wyniki na x0, y0 mają następującą postać:

$x0 = 0.5 * (x1+x2)$
$y0 = 0.5 * (y1+y2)$

W tym wypadku środek okręgu leży w środku odcinka AB. Jest to ilustracja kolejnego twierdzenia z geometrii, że kąt wpisany oparty na półokręgu jest kątem prostym.

Promień okręgu wyznaczony jako odległość między punktem x0, y0, a punktem x1, y1.

$$r = \sqrt{(x1-x0)^2 + (y1-y0)^2}$$

PROGRAMOWANIE

Na listingu 1 przedstawiono procedurę *CircleFrom3Points* będącą pascalową implementacją przedstawionych wzorów. Jej trzy pierwsze parametry x0, y0, r, zadeklarowane przez VAR, odpowiadają współrzędnym środka okręgu i jego promieniowi. Sześć następnych x1,y1, x2,y2, x3,y3 — to współrzędne trzech kolejnych punktów A, B, C. Jeśli wartość zmiennej sDen, typu REAL, jest bliska zeru, to wykonanie programu zostanie przerwane. Warunek ten odpowiada współliniowości punktów A, B, C.

Na listingu 2 przedstawiono program demonstracyjny C3P–DEMO. Jego część główna składa się z wywołań trzech procedur: *GetPoints, CircleFrom3Points i PrintCircleParameters*. Pierwsza procedura pozwala na wprowadzenie współrzędnych trzech punktów trójkąta, natomiast trzecia wyprowadza na ekran komputera wyniki: x0, y0 i r. Druga procedura znajduje się w zbiorze C3P.PAS, dołączonym instrukcją INCLUDE, i została omówiona poprzednio.

Przedstawiony program ma budowę modularną, z wyodrębnionymi częściami opowiedzialnymi za wprowadzanie danych, ich przetwarzanie i wyprowadzanie otrzymanych wyników. Mimo jednokrotnego użycia, niektóre fragmenty kodu zastąpiono procedurami. Pozwala to na łatwiejsze zrozumienie i modyfikacje tak napisanych programów, zwłaszcza, gdy analizowany problem jest bardziej skomplikowany, niż ten który przedstawiliśmy w tym artykule.

Jonasz Mayer

```
procedure CircleFrom3Points ( var x0,y0, r : real;
                               x1,y1, x2,y2, x3,y3 : real );
{****************************************************}
{ procedura wyznacza z trzech punktow x1,y1, x2,y2, }
{ x3,y3 wspolrzedne srodka okregu x0,y0 oraz jego   }
{ promien r.     (C) JM Maj 1989                     }
{****************************************************}
var
   s, sNum, sDen : real;
begin
    sNum := (y3-y1) * (y3-y2) + (x3-x2) * (x3-x1);
    sDen := (x2-x1) * (y3-y2) - (x3-x2) * (y2-y1);
    if abs(sDen)<=0.000001 then halt;
    s := sNum / sDen;
    x0 := 0.5 * ( (x1+x2) - (y2-y1) * s );
    y0 := 0.5 * ( (y1+y2) + (x2-x1) * s );
    r := SQRT ( SQR (x1-x0) + SQR (y1-y0) );
end; { Circle from 3 points }
```

Listing 1. Zbior C3P.PAS

```
Program C3P_DEMO;

{$I C3P.PAS }

var
   x0,y0, r : real;
   x1,y1, x2,y2, x3,y3 : real;

procedure GetPoints (var x1,y1, x2,y2, x3,y3 : real);
begin
  writeln(' Wyznaczanie parametrow okregu ',
          'okreslonego przez 3 punkty');
  write ('punkt 1 (x y): ');  readln (x1,y1);
  write ('punkt 2 (x y): ');  readln (x2,y2);
  write ('punkt 3 (x y): ');  readln (x3,y3);
end; { GetPoints }

procedure PrintCircleParameters (x0,y0, r : real);
begin
  writeln ('parametry okregu: ','x0=',x0:10:4,
           ' y0=',y0:10:4,' r=',r:10:4);
end; { Print Circle Parameters }

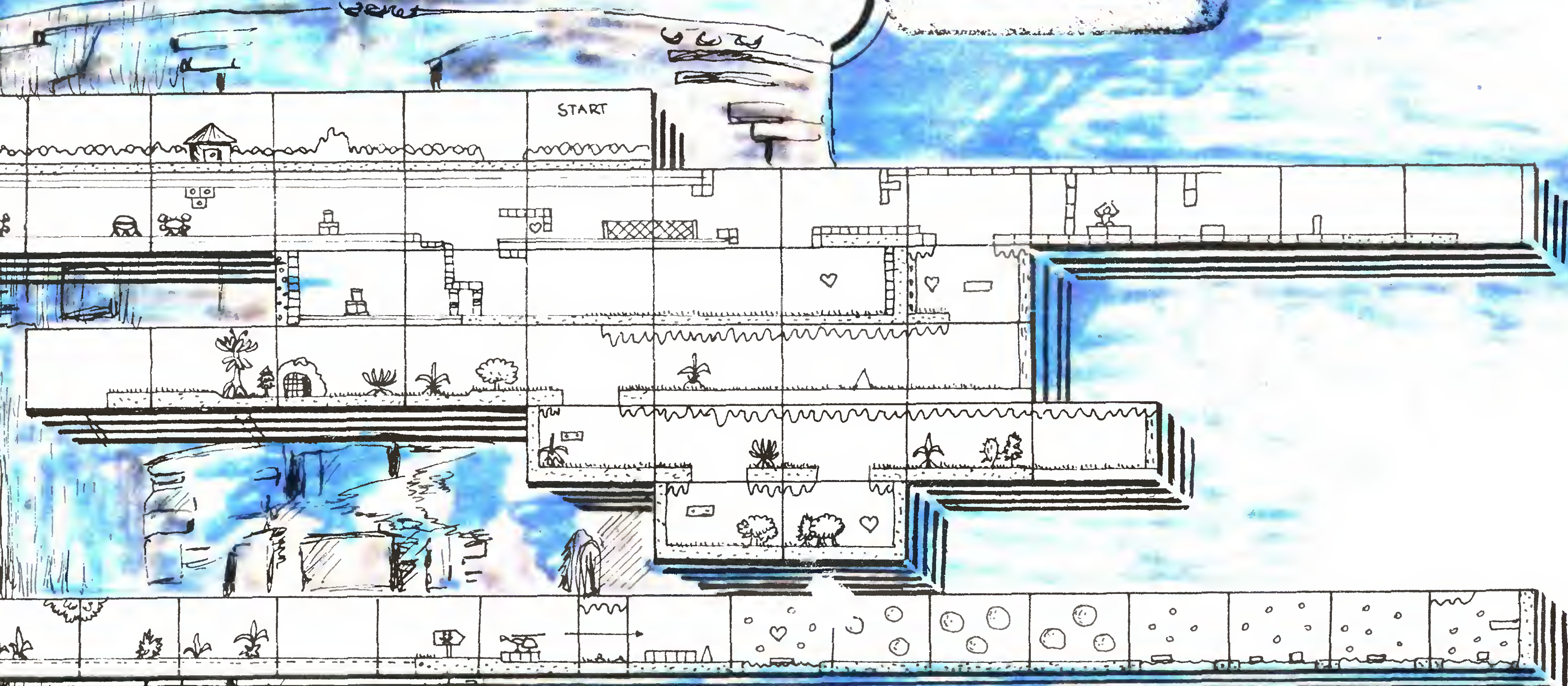
begin
    GetPoints (x1,y1, x2,y2, x3,y3);
    CircleFrom3Points (x0,y0,r, x1,y1, x2,y2, x3,y3);
    PrintCircleParameters (x0,y0, r);
end.
```

Listing 2. Zbior C3PDEMO.PAS.



GAME







„Gremla, Twoja gra skończona! Wyłaź, rozwalę Cię jak szczura!" — zakrzyknął krwiożerczy Arkos, gdy dopadł swego śmiertelnego wroga w skalnej pułapce na planecie Dolphas. Gonił go przez całą galaktykę i wreszcie go ma! Zemsta będzie krwawa. Niekończące się lata tyranii wywarły głębokie piętno na charakterze Arkosa, przywódcy rebelii. Nie będzie miał on litości dla swego ciemięzcy...

Nie było jednak pisane, by zemsta dopełniła się teraz. Z nieba błyskawicznie spadł bojowy oddział istot zwanych Ardec i obezwładnił Arkosa. Gremla natychmiast wyszedł z ukrycia, kazał spętać Arkosa i odesłać go na więzienną planetoidę Phantis.

Majorze Locke! Wszyscy znamy Twe męstwo. Wierzymy, że zdolny jesteś do największych poświęceń. Czy pozostawisz Arkosa na pastwę okrutnego Gremli? Czy pozwolisz, by umarł on w mękach, a jego lud wyginął bez śladu? Na pewno nie! Wyruszaj więc, każda chwila jest droga.

Locke nie wahał się długo. Prawie natychmiast wsiadł do międzygalaktycznego transportowca, zdążającego w kierunku najczarniejszych otchłani Kosmosu, gdzie znajduje się planetoida Phantis. Po dwóch godzinach żmudnych obliczeń znał już moment, w którym powinien oderwać się swym małym pojazdem od transportowca.

Gdy mała, świecąca wskazówka pokryła się z czerwonym punktem na tarczy chronometru Locke'a, ujął on dźwignię przyspieszenia i szarpnął do siebie. Niewielki myśliwiec, w sam raz do takich wypadów, zadrżał, buchnął ogniem i pomknął ku widniejącej w czerni brązowej Phantis.

Rosła ona coraz bardziej i po chwili wypełniła cały monitor myśliwca. System autonawigacji zamrugał kolorowymi lampkami i statek wleciał w atmosferę planetoidy — celu swej podróży.

Atomy pierwiastków tworzących tę atmosferę z błyskiem jonizowały się na powierzchni pojazdu, który ciął ją zmierzając ku powierzchni. Już tylko 700 mil.

Nagle do błysków tych dołączyły inne — wybuchy pocisków, które rozrywały się niebezpiecznie blisko myśliwca sterowanego przez Locke'a. Jednocześnie nadleciały dwie bojowe eskadry pojazdów Gremli. Lądowanie nie było łatwe — wrogie obiekty, choć rozrywały się po jednym strzale, skutecznie przeszkadzały w utrzymaniu kursu. Ale Locke, zaprawiony w bojach marines dał sobie radę i szczęśliwie wylądował.

Gdyby wiedział, co go tam czeka, zastanowiłby się, czy warto lądować w samym sercu bazy nieprzyjaciela. Minusów było sporo, zaś plus tylko jeden — baza była pusta. Jeśli oczywiście nie liczyć małego Ardeca, który po przelutowaniu dwóch oporników jak pies podążył za Locke'em.

Wędrując odpierali ataki żołnierzy Gremli, mało skuteczne, lecz mimo wszystko uciążliwe. Wejście do podziemi zostało zdobyte szturmem, broniący rozpierzchli się jak muchy.

Już na pierwszym poziomie podziemi Locke znalazł dziwny pistolet. Czy to nie pułapka? Nie było czasu myśleć. Wybuchający korpus nadbiegającego stwora potwierdził użyteczność pistoletu. Ardec został na zewnątrz, więc znalezisko będzie tym cenniejsze.

Po dokładniejszym rozejrzeniu się wokoło Locke znalazł magazynek protopocisków do pistoletu. Tak uzbrojony wędrował podziemnymi korytarzami, jeśli wędrowaniem nazwać można ciągłe składanie się do strzału.

Podziemne jezioro, dziwne pnącza i konstrukcje, wreszcie zakratowane wejście do więzień — to wszystko napotkał nieustraszony Locke w czeluściach Phantis, o której krążą legendy od wieków. On też miał stać się częścią jednej z nich.

Znaleziony medalion z napisem w nieznanym języku był bardzo ciężki. Mimo to Locke niósł go, gdyż wierzył, że przeczucie nie myli go. I nie zawiódł się. Wystarczyło podejść do kraty, gdy usunęła się ona z niegłośnym trzaskiem. Medalion był niepotrzebny, zaś bohatera czekała ostateczna rozgrywka.

Więzienia były wilgotne i ponure. Rowy z kipiącą cieczą i spadające bloki skalne przeszkadzały w wędrówce, lecz nie zdołały powstrzymać człowieka o nieustraszonym sercu. Tylko ten, w którego żyłach płynie krew pogromców kosmicznych potworów ma szansę na przebycie straszliwych lochów. Locke dokonał tego.

W ostatniej celi leżał przykuty do ściany Arkos. Pobyt w więzieniu wywarł widoczne piętno na jego ciele i duszy. Mimo to instynktownie poczuł zbliżanie się przyjaciela. Ten uwolnił go jednym ruchem plazmowego noża. Potem błyskawicznie wydostali się na powierzchnię, wsiedli do największego z pojazdów Gremli i odlecieli do domu, by nigdy tu nie powrócić.

„Co to za błysk, tam za nami?" — spytał Arkos, gdy byli już daleko. „A, zapomniałem Ci powiedzieć. To Phantis zamieniła się w obłok pyłu, który rozsypie się po całej galaktyce. Zostawiłem Gremli niespodziankę w magazynach paliwa do statków kosmicznych".

Firma: Dinamic
Komputer: ZX-Spectrum 48/+, Commodore 64, Amstrad CPC, Atari ST, Amiga, IBM PC, MSX

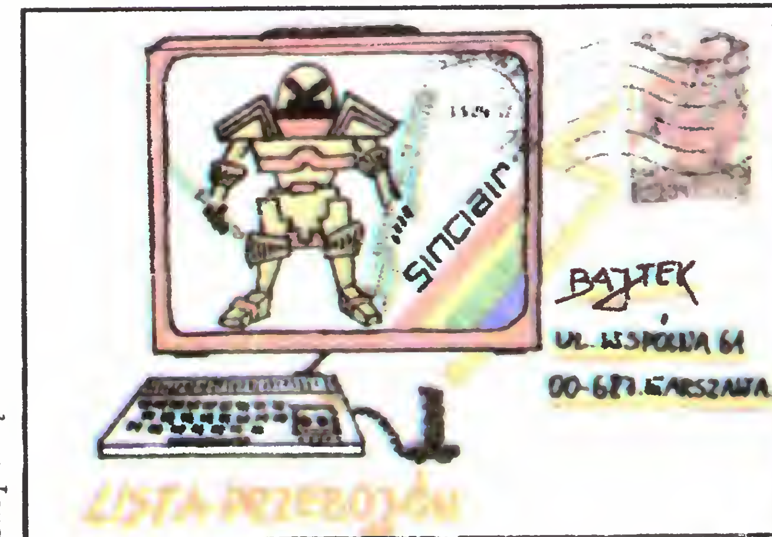
Gen

10 BAJTKOWA LISTA PRZEBOJÓW (9/89)

Cóż za siła! Operation Wolf wciąż króluje! I jak widać z listów, szybko nie spadnie. Tuż za nim RoboCop, który ma jak dotąd największe szanse do wygryzienia Operation Wolfa. Cybernoid, choć spadł, to jednak potrzyma się jeszcze. Choć tę pierwszą trójkę atakuje Captain Blood i Traz, na razie nic nie zapowiada rewelacji.

Na wrześniowe notowanie otrzymaliśmy 4061 propozycji na 198 tytułów gier.

			ATARI	AMSTRAD	COMMODORE	SPECTRUM
1	OPERATION WOLF		>	<	×	×
2	ROBOCOP	!	>	<	×	×
3	CYBERNOID	↓	>	<	<	×
4	CAPTAIN BLOOD	↑	>	×	×	<
5	ROCKET RANGER	↓	>		<	
6	TRAZ	!	<	<	<	<
7	BMX+	↓	<	<	×	×
8	GUNSHIP	↓	>	×	×	<
9	STREET FIGHTER	↓	>	<	×	<
10	HOT SHOT	↓	>	<	<	×



Kopertę nadesłał Tomasz Wróblewski z Wrocławia.



BOB'N RUMBLE

Pierwszą jego myślą po przebudzeniu było to, że jest w Alcatraz. Drugą, że chętnie wyssałby tego, kto zrobił pryczę, na której męczył się całą noc, ostatnia zaś dotyczyła ohydnej potrawy, której żołądek z pewnością długo jeszcze nie strawi.

Bob nie był jednak człowiekiem umysłowym, wręcz przeciwnie — należał do typu fizycznych z zerowym współczynnikiem IQ. Szybko wstał, zmierzwił włosy, rozgniótł obcasem leżącego kota i wyszedł z domu (bardzo ładne określenie na tę lepiankę).

Po ulicy ciągnął tłum przechodniów — szarych ludzi, wszystkich podobnych i takich samych. Jakiś reporter przeprowadzał wywiad z uczestnikami konkursu „Wiedziałem wszystko". Bob kątem ucha usłyszał urywek zdania: „... tak, rozumie Pan. ... ttto jest dla nas ttaka wielka przygoda... wwielka, i chciałem jeszcze dodać od siebie, że my to traktujemy jjako taką przygodę po prostu..." Reszta zabawnego monologu zagłuszona została przez brutalny wrzask wpadającego w kałużę bachora.

W nudnym i monotonnym tłumie ulicznym były osoby stanowczo ciekawsze. Oto dwóch młodzianów przed sklepem, których kurtki wypchane były w kształtach wykluczających możliwość, że mają tam bułki z masłem. Po parku biegał zwariowany emeryt w okularach wymachując wielkim tureckim mieczem. Jakiś wampirowaty dzieciak kopał swoją rodzicielkę mówiąc: „mamo, kup mi tę cegłę". Bob jednak nie zwracał na nikogo większej uwagi, lecz szedł prosto do sklepu monopolowego.

Nasz bohater bardzo nie lubił, gdy ktoś go zaczepiał na ulicy. Zwykle agent taki lądował z głową w kanale ściekowym, lub zbierał zęby w woreczek (jeśli był w stanie je dojrzeć). Dziś jednak Bob był bardzo spokojny — gdy jeden z przechodniów zaproponował mu prostowanie bananów — tylko kopnął go z pogardą. Mimo to człowiek ten już nigdy nie podniósł się z ziemi o własnych siłach. Na harmider wywołany dźwiękiem, jaki wydał upadający denat, zbiegło się kilku dryblasów gotowych zrobić zadymę. I właśnie w tej chwili usiadłeś do komputera i zacząłeś grać.

Bob jest człowiekiem (to chyba zbyt śmiały wniosek), który nauczył się bić na różne sposoby. Może więc kopać, boksować się, miażdżyć przeciwnika, nacierać mu uszy itp. Jego kartą atutową jest jednak obrót dookoła głowy i rzut skazańcem w najbliższy mur. Potrafi on także złapać za kurtkę i jednym uderzeniem „czachy o czachę" pozbawić tchu. Widać więc dobrze, że nie jest to gra dla tych, którzy marszczą się po każdym przejawie siły na ekranie — wręcz przeciwnie. Najlepszymi graczami są bezwzględni mordercy, mający możliwość wyładować się nie raniąc prawdziwych ludzi.

W każdym z ośmiu etapów masz do czynienia z innymi przeciwnikami, zmieniają się też ciosy, jakich możesz używać. Duszek pojawiający się od czasu do czasu przynosi serce lub laskę dynamitu. Serce uzupełni Twój zapas energii, bomba zaś uszczupli ją, więc szybko ją odrzuć. Pamiętaj też, że jedynym sposobem na buldoga jest połachotać go w brzuch, a natrętnego emeryta uspokoi tylko mocny cios w te jego obleśne okulary.

Firma: Beam Software
Komputer: Commodore 64, ZX Spectrum

Luke

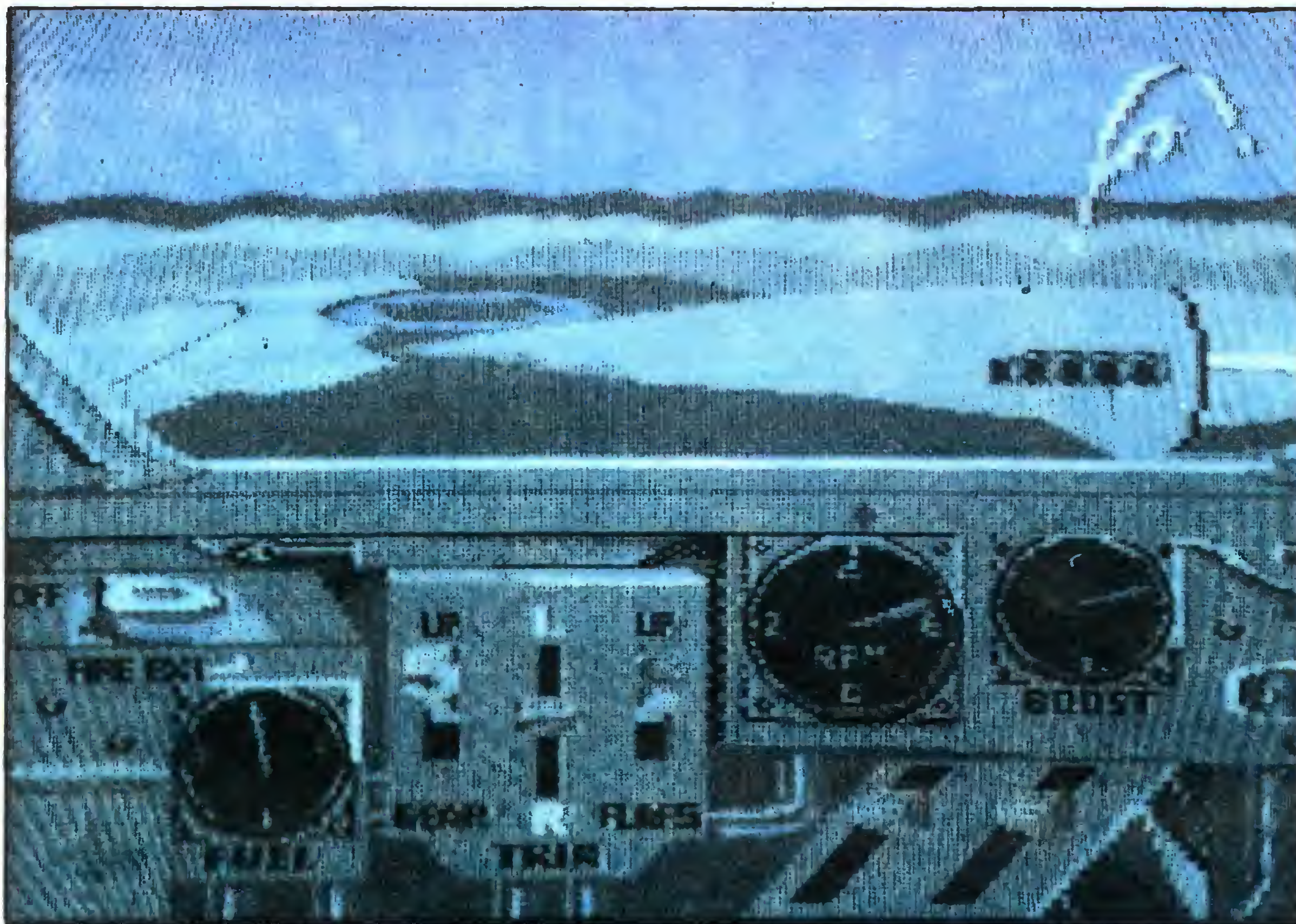
KRÓL I KRÓLOWA GIER



Małgosia Krasoń, lat 8. II klasa Szkoły Podstawowej nr 44 we Wrocławiu. Ulubiony komputer: Junior. Gry: Pinball, Arkanoid. Zainteresowania: zbieranie pocztówek.



Łukasz Łabuś, lat 11. V klasa Szkoły Podstawowej nr 75 we Wrocławiu. Ulubiony komputer: Junior. Gry: Hero, Frankenstein. Inne zainteresowania: modelarstwo.



ACE OF ACES

Usiadłem za sterami i zapiąłem pasy bezpieczeństwa. Plan lotu, mapę Francji i prognozę „meteo" rzuciłem na stolik Jessa — pokładowego nawigatora. W rękę ująłem metalowy drążek sterowy, nogi położyłem na pedałach. Jeszcze wcisnąć kilka przycisków, sprawdzić radio, stany liczników i można zapuszczać silniki. Potem już tylko zwolnić hamulce i wykołować na miejsce, gdzie po otrzymaniu pozwolenia na start, wyjące na pełnych obrotach silniki wyniosą Cię w powietrze.

Mosquito — doskonały brytyjski samolot, podczas Drugiej Wojny Światowej występował w wielu rolach — jako bombowiec, myśliwiec i nocny myśliwiec. Wersja seryjna uzbrojona była tylko w czetry karabiny maszynowe 20 mm, lecz zdarzały się maszyny posiadające 6 działek o kalibrze 57 mm. W luku bombowym mógł pomieścić do 2000 kg bomb stu- lub dwustukilogramowych. Nie brakło także miejsca na rakiety przeciwlotnicze, jeszcze nie samonaprowadzające, ale mimo to wystarczająco celne, aby każdy niemiecki samolot zamienić w obłok dymu. Najciekawsze jest jednak to, że ten samolot, używany do najtrudniejszych misji specjalnych w pięćdziesięciu procentach składał się z płótna i dykty.

Oczywiście, jedynym sposobem aby wystartować na tym bezpiecznym angielskim myśliwcu, jest załadowanie gry „ACE OF ACES". Tylko Ci gracze, którzy spędzili miesiące na rozpracowywaniu licznych symulatorów lotu, mogą bez żadnych niespodzianek zagrać w tą grę. Innym radzę odkurzyć stare, zapomniane już taśmy i poćwiczyć na wszystkich znalezionych tam symulacjach.

Na początku mamy do wyboru cztery różne scenariusze niszczenia rakiet V1 lecących na Londyn, walka z Junkersami podążającymi w tym samym kierunku, zatrzymanie pociągu wojskowego (tak aby już został tam na zawsze), zbombardowanie bazy U-bootów. Gracz ma możliwość walki na wszystkich czterech scenariuszach naraz, oraz praktyki. Po wybraniu rodzaju walki, otrzymujesz dokładny plan lotu z podanymi wysokościami i kierunkami lotów celów. Nie zapomniano także o komunikacie pogody, oraz mapie z zaznaczonymi punktami wyjściowymi.

Poszczególne miejsca w samolocie zmienia się klawiszami **1—5**. Podaję funkcję każdego z nich, oraz wskaźniki znajdujące się w niektórych kabinach:

1 — kabina pilota (od lewej)
— kompas,
— licznik prędkości,
— sztuczny horyzont,
— wskaźnik wysokości,
— radar,
— wskaźnik wysokości samolotu wroga
2 — lewy silnik
3 — prawy silnik
— FIRE EXT (gaśnica),
— FUEL (licznik paliwa),
— TRIM (przeciwwaga, używaj gdy jeden z silników nie działa),
— FLAPS (klapy góra — dół),
— GEAR (podwozie góra — dół),
— RPM (licznik obrotów silnika),
— BOOTS (kąt natarcia łopat śmigieł),
4 — mapa,
5 — luk bombowy,
— FUEL TANK (włącznik rezerwy paliwa),
— OPEN-CLOSE (luk otwarty-zamknięty),
— GUN ROCKET (przełącznik działko-rakiety).

Gra ta znakomicie oddaje wszystkie prawie tajniki lotu Mosquitem. Charakterystyczne palenie się silników, blokujące podwozie, drgania i wibracje to tylko niektóre z „atrakcji" przeniesionych z prawdziwej maszyny. Czasami nie otwierająca się osłona w luku bombowym uniemożliwia zrzucenie najmniejszej nawet bomby, niebezpieczniejsze są jednak myśliwce wroga, które kończą Twoją misję po kilku celnych seriach.

Program wyróżnia się pod względem graficznym, ma także doskonałe efekty specjalne. Dźwięki strzelającego Messerschmitta, spadającej bomby, wybuch wagonu z powodzeniem mogą zmylić niejednego znawcę. Myślę, że jest to jeden z najlepszych symulatorów samolotu myśliwsko-bombowego i z tego względu zasługuje na dużą popularność.

Firma: Accolade
Komputer: ZX Spectrum 48/+, Commodore 64, Atari XL/XE

LUKE

ROZSTRZYGNIĘCIE KONKURSU „TYLKO DLA GRACZY"

Ponieważ „Bajtek" nr 4 ukazał się w kioskach ok. 3 dni przed upłynięciem terminu nadsyłania rozwiązań, przeto rozwiązania te przyjmowaliśmy do 31 czerwca.

Nadeszło ich całe pudło: ponad 1500 kartek pocztowych i listów. Większość zawierała rozwiązanie poprawne, tzn. hasło „gry to nie wszystko", mimo to zdarzały się improwizacje zaskakujące swą oryginalnością. Nie mówiąc o propozycjach typu „Wesołych Świąt" lub „Czytaj Bajtka", zdarzały się nawet „Niech żyje Atari".

Jakie tytuły gier złożyły się na hasło? Oto one: RENEGADE, PREDATOR, BUGGY BOY, TRANTOR, EXOLON, DAN DARE II, ACTION FORCE II, VIXEN, T-WRECKS, GUNSMOKE, SKATE GRAZY, MICKEY MOUSE, FLINTSTONES, STREET FIGHTER, VENOM STRIKES BACK i PLATOON.

Spośród nadesłanych rozwiązań wylosowaliśmy dziesięć listów. Joysticki otrzymali: **Marcin Brodowski z Tych, Przemysław Garbacz z Kielc, Paweł Wojciechowski z Lublina, Jacek Spruch z Krotoszyna i Tomasz Kiraga z Bielsko-Białej.** Natomiast kasety z programami otrzymają: **Krzysztof Grosicki z Kielc, Rafał Patzur z Warszawy, Marcin Pankowski z Lubartowa, Przemysław Trafalski z Łodzi i Michał Wilanowicz,** którego prosimy o kontakt, gdyż adresu nie podał. Nagrodę specjalną — numer miesięcznika Sinclair User otrzymuje Grześ Bucholc z Katowic.

Nagrody wyślemy pocztą.

S.O.S.

Mam 11 lat i jestem posiadaczką Atari 800XL. Nie wiem, jak zacząć grę EASTERN FRONT. Proszę także o pomoc w grze Zorro — dochodzę do lochów i nie wiem, gdzie są drzwi z butem, kielichem i podkową.
Agata Violoncellista, ul. Pocztowa 18, 05-520 Konstancin-Jeziorna

Czy ktoś z Czytelników „Bajtka" może podać mi opis gry „Wizard's Lair"?
Daniel Gumowski, ul. Szareckiego 1/23, Warszawa.

Pomocy! O co chodzi w grach: HUNT FOR RED OCTOBER, SILENT SERVICE, GUNSHIP, TRADING PSI 5 COMPANY? Bardzo proszę o pomoc!
Tomasz Aniszewski, ul. Króla Kazimierza 12, 04-854 Radość

Poszukuję opisu do gry COLONY i kodu uruchamiającego do gry BARBARIAN w wersji na Atari.
Adam Pisarek, ul. Połomska 5a, 44-266 Świerklany

Od dłuższego czasu szukam opisów do gier BOOTY, POP EYE, GHOSTS'N GOBLINS, PHANTOMAS. Będę bardzo wdzięczna za pomoc.
Katarzyna Lipińska, ul. Paganiniego 12//65, 20-854 Lublin

Mieszkam na wsi, do której przydzielony został mikrokomputer TIMEX 2048. Poza umiejętnością grania niewiele umiem na nim zrobić. Chciałbym więc nawiązać kontakt z posiadaczami tego mikrokomputera; myślę iż wiele mógłbym się nauczyć.
Tomasz Bielawski, Łęgowo, 82-560 Kisielice, woj. Elbląg

Poszukuję następujących gier na Commodore 64: THE TRAIN, MOUSE TRAP, SUBMARINE COMMANDER, GREEN BERET, KING'S QUEST. W zamian oferuję np. TURBO 2002, PLATOON, BARBARIAN (I i II), RENEGADE, RYGAR, ACE II.
Piotr Miczek, ul. Starobielska 1/6, 43-300 Bielsko-Biała

Proszę o pomoc w grach HACKER, NEW YORK CITY, JUMBO JET PILOT, AZTEC. W zamian służę informacjami o wielu grach.
Grzegorz Schmidt, ul. Obr. Stalingradu 125b/12, 40-616 Katowice

Mam COMMODORE16. Pilnie poszukuję dokładnych opisów i map do gier COBS and ROBERTS i OPERATION RED MOON.
Dawid Bębenek, ul. Rożdzieńskiego 86//185, 400-950 Kielce

Liczę na waszą pomoc w zdobyciu gier RAMBO, SPY HUNTER, GHOSTS'N GOBLINS, SABOTEUR II, JACK THE NIPPER na SPECTRUM. W zamian oddam ZORRO, DAM BUSTERS, MOVIE, MATH DAY i inne.
Tomasz Nowak, ul. Filtrowa 29b, 87-100 Toruń

Bardzo proszę o klawisze sterujące do gry GRAND NATIONAL na SPECTRUM.
Tomasz Francuz, ul. Pułaskiego 3/7, 41-800 Zabrze

Pilnie poszukuję gry GHOSTS'N GOBLINS na Atari 800XL. Posiadam ok. 250 gier. Za wyżej wymienioną grę odstąpię 15 dowolnie wybranych.
Paweł Bąk, ul. Kalcytowa 1/55, 25-705 Kielce

Uprzejmie proszę o przysłanie mi loadera dyskowego do gry MIKIE w wersji na AMSTRADA CPC 6128.
Michał Maciejowski, os. 25-lecia PRL 23/10, 58-260 Bielawa, woj. wałbrzyskie

Pomocy! Potrzebuję nieśmiertelności do gier MISTER ROBOT, MONTEZUMA'S REVENGE i RIVER RAID. Pomóżcie! Posiadam Atari 65XE.
Zuzanna Antoniewicz, ul. Mikołaja Reja 16/1, 47-220 Kędzierzyn

Poszukuję dokładnego opisu gry KENNEDY APPROACH na Atari 800XL. Liczę na waszą pomoc.
Andrzej Dębicki, ul. Solarza 9/4, 35-111 Rzeszów

Proszę o dokładny opis do gier TOMAHAWK, KING OF THE RING, SOLO FLIGHT, THE NINJA MASTER. Posiadam Atari 800XL.
Maciej Meisler, ul. Świnoujska 5a, 54-313 Wrocław



KOPROCESOR ARYTMETYCZNY

Chociaż większość naszych czytelników używa małych komputerów ośmiobitowych, i nie zanosi się na to by coś się w tej sprawie zmieniło, postanowiłem napisać kilka słów o koprocesorach arytmetycznych, będących rozwiązaniem stosowanym w systemach szesnastobitowych. Myślę że wszyscy poważni użytkownicy mikrokomputerów chętnie dowiedzą się czegoś nowego — może kiedyś powstanie koprocesor do Spectrum? Demon szybkości i magia Mega (herców, bajtów itd.) działa na prawie wszystkich. A więc...

CO I PO CO?

Jak wiadomo, w wielu zastosowaniach komputerów potrzebne jest wykonywanie obliczeń zmiennoprzecinkowych (pisaliśmy już o nich kilka razy). Wszystkie programy takie jak arkusze kalkulacyjne, symulatory (gry i nie tylko gry) czy programy służące do prowadzenia księgowości, muszą wykonywać obliczenia na liczbach o różnej precyzji. Tylko programy którym obliczenia są potrzebne jedynie dla zapewnienia porządku w nich samych (i komputerze), takie jak systemy operacyjne czy edytory tekstów, wykonują obliczenia wyłącznie na liczbach całkowitych.

Ponieważ mikroprocesor, stanowiący serce komputera nie potrafi sam wykonywać obliczeń zmiennoprzecinkowych, musi w tym celu wykorzystywać skomplikowane podprogramy. Dobrze napisany podprogram, mnożący przez siebie dwie liczby zmiennoprzecinkowe na komputerze PC z procesorem 8088 potrzebuje na wykonanie tej operacji około 900 mikrosekund. Mowa tu o liczbach dużej precyzji czyli o 18-stu cyfrach znaczących. Większość programów nie dysponuje takimi możliwościami i obliczenia są wykonywane na liczbach o precyzji 8 cyfr, czyli takich samych jak w Spectrum. Jest to bardzo szybko w porównaniu z ręcznym kalkulatorem, ale duże programy takie jak ACAD muszą często wykonywać setki tysięcy, jeśli nie miliony takich operacji, żeby przygotować jeden rysunek. I tu pojawia się miejsce dla koprocesora.

W każdym (albo prawie każdym) komputerze klasy IBM PC na płycie głównej znajduje się gniazdo na koprocesor arytmetyczny. Koprocesor to taki dodatkowy układ scalony, którego jedynym przeznaczeniem jest wykonywanie obliczeń zmiennoprzecinkowych nie przy pomocy podprogramu, ale dzięki odpowiedniej konstrukcji — jeżeli zwykły procesor może wykonywać obliczenia na liczbach całkowitych, to odpowiednio zaprojektowany inny układ może wykonywać takie same obliczenia na liczbach zmiennoprzecinkowych. Koprocesory firmy Intel wykonują identyczne mnożenie jak to, o którym była mowa powyżej, w ciągu 20 do 30 mikrosekund, czyli ponad trzydzieści razy szybciej.

JAK TO DZIAŁA?

Koprocesor ma za zadanie zwiększyć możliwości procesora nie przeszkadzając w jego normalnej pracy. W związku z tym zdecydowano się na następujące rozwiązanie — do sterowania koprocesorem istnieją specjalne instrukcje, które w programie współpracującym z koprocesorem przeplatają się z normalnymi instrukcjami dla głównego procesora. W przypadku braku koprocesora instrukcje dla niego są po prostu ignorowane. Oznacza to niestety że w przypadku braku koprocesora niektóre programy mogą nie nadawać się do użytku. Na szczęście można je tak napisać, żeby były w stanie samodzielnie sprawdzić czy koprocesor jest czy go nie ma, i w zależności od tego korzystać zeń lub nie. Program który nie zawiera instrukcji dla koprocesora nic na jego obecności nie skorzysta — będą się nawzajem ignorować, bez żadnych zysków ani strat dla obu stron.

JAKIE ZYSKI?

Że pojedyncze obliczenia są mniej więcej trzydzieści razy szybsze już napisałem. Oznacza to, że PC może w ciągu jednej sekundy wykonać około 100 do 200 tysięcy operacji zmiennoprzecinkowych (zależnie od modelu i częstotliwości zegara). Zastosowany format liczby to 64 bity na cechę, 15 bitów na matysę i jeden bit znaku. W sumie — 10 bajtów, czyli 5 słów. Oznacza to w praktyce 10 cyfr znaczących i niewyobrażalny zakres liczb od $1*10^{-4931}$ do $1 tt 1^{+4931}$. Zarówno zakres i precyzja tych liczb przekraczają te wartości, które znane są z wielu dużych komputerów (oczywiście nie ze wszystkich). Dobry IBM PC z koprocesorem potrafi osiągnąć podczas przeprowadzenia obliczeń szybkość rzędu 1/2 do 1/20 szybkości dużego komputera, a kosztuje 10 do 100 razy mniej.

Najbardziej znaną rodziną koprocesorów są wspomagające procesory rodziny 80*86 koprocesory firmy Intel. Pierwszy z trzech braci,

INTEL 8087

był pierwszym koprocesorem do PC. Można go użyć w komputerach wyposażonych w procesor 8088 albo 8086 (czyli z grubsza rzecz biorąc IBM PC XT i kompatybilne). Koprocesor i procesor są połączone razem „noga w nogę" i stanowią zespół który można traktować jak jeden procesor uniwersalny. Takie rozwiązanie powoduje że obydwa korzystają z zegara o tej samej częstotliwości. W efekcie w pierwszych wersjach koprocesor jako znacznie bardziej skomplikowany układ przy dużych (wówczas) częstotliwościach zegara taktującego (czyli w okolicach 8 MHz) zaczynał się łatwo przegrzewać i powodował błędy w obliczeniach, a nawet potrafił zdezorganizować pracę całego komputera i spowodować jego zawieszenie. Zanim technologia poszła na tyle do przodu, że udało się skonstruować układy nie przegrzewające się tak łatwo, firma Intel zaproponowała inne rozwiązanie w koprocesorze,

INTEL 80287.

Przystosowany był on do pracy z procesorem 80286 (lub z wczesnymi wersjami procesora 80386) stosowanym w komputerze IBM PC AT. W odróżnieniu od koprocesora 8087 pracuje on w sposób asynchroniczny, to znaczy że jest taktowany inną częstotliwością zegara niż główny procesor (dwie trzecie jego częstotliwości). Powoduje to pewne problemy przy komunikacji między procesorem i koprocesorem — w efekcie taki zestaw jest mniej efektywny niż 8088/8087, za to można używać wolniejszego koprocesora do szybkiego procesora, co miało kilka lat temu niemałe znaczenie. Porównanie obu koprocesorów — 8087 i 80287 — przy tej samej częstotliwości zegara wypada na korzyść 8087, który jest (dzięki znacznie lepiej przybiegającej współpracy z procesorem) o mniej więcej 20% szybszy.

Ostatni z rodziny

INTEL 80387

jest przeznaczony do pracy z procesorem 80386. Jest on w pełni kompatybilny z 8087 i 80287, ale tylko w jedną stronę — to znaczy, że przy jego pomocy można wykonywać każdy program napisany na wcześniejsze modele, ale w przeciwnym kierunku programów przenosić się nie da. 80387 jest bowiem wyposażony w dodatkowe instrukcje, pozwalające na obliczanie wszystkich funkcji trygonometrycznych — niektórymi z nich jego młodsi bracia nie dysponują. Wprawdzie koprocesor ten jest również koprocesorem asynchronicznym, ale po pierwsze na ogół jest taktowany tą samą częstotliwością co główny procesor (20 albo 25 MHz — technologia nie stoi w miejscu...), po drugie dzięki dokonaniu pewnej optymalizacji operacji komunikowania się z procesorem 80386 straty czasu potrzebnego na wymianę informacji są znacznie mniejsze.

SZYBCIEJ CZY WOLNIEJ?

Zmniejszona wydajność koprocesora 80287 powoduje pewne nieoczekiwane efekty podczas porównywania komputerów klasy AT, wyposażonych w zestaw 80286/ /80287 z komputerami XT, opartymi na kombinacji 8088/8087 (lub 8086/8087). Po pierwsze, żeby móc się równać w szybkości obliczeń z XT z zegarem 10 MHz, AT musi dysponować zegarem 14 MHz (normalnie przy tej samej częstotliwości zegara AT jest około trzech razy szybsze od XT), po drugie, niektóre programy korzystające z koprocesora mogą być na komputerach AT wolniejsze przy jego użyciu niż bez niego. Spowodowane jest to tym, że procesor 80286 jest bardzo szybki podczas wykonywania obliczeń na liczbach całkowitych (a potrafi je w odróżnieniu od procesorów ośmiobitowych również mnożyć i dzielić). Nie najlepiej napisany program, korzystający podczas obliczeń na liczbach całkowitych z koprocesora 80287 może w trakcie komunikowania się procesora z koprocesorem stracić tyle czasu, że w efekcie będzie działał wolniej.

O ILE SZYBCIEJ Z KOPROCESOREM?

Jest to najczęściej stawiane pytanie dotyczące koprocesorów. Nie jest na nie z różnych powodów łatwo odpowiedzieć. Wszystko zależy od tego jak program jest napisany, i ile czasu zajmuje w nim wykonywanie obliczeń. Jeżeli w programie 50% czasu potrzebne jest zarządzanie pamięcią i wykonywanie instrukcji nie związanych z obliczeniami zmiennoprzecinkowymi, przyspieszeniu w obecności koprocesora może ulec tylko pozostałe 50%, i to pod warunkiem że program jest w stanie koprocesor wykorzystać. Różnica może być trzydziestokrotna, a może jej nie być wcale — jedynym sposobem by się o tym przekonać jest wykonanie testu — trzeba program uruchomić na komputerze z koprocesorem i bez, mierząc czas potrzebny do jego wykonania. Innego sposobu nie ma.

MICROWAY

to firma o której trzeba w tym miejscu wspomnieć — z kilku powodów. Po pierwsze niniejszy tekst powstał dzięki materiałom nadesłanym przez przedstawicielstwo firmy w Europie (MicroWay jest spółką amerykańską, ale działa na całym świecie), za co serdecznie dziękuję. Po drugie MicroWay ma duże zasługi dla użytkowników PC — od 1981 roku zajmuje się przyspieszaniem obliczeń wykonywanych na komputerach osobistych — przy pomocy koprocesorów (nie tylko opisanej powyżej rodziny Intel-a, ale również innych firm — np. Inmos, Weitek, Motorola), transputerów i innych rozwiązań — np. urządzenia o nazwie MicroWay 287 Turbo, służącego do zwiększania efektywności koprocesora 80287. Czy teraz wiecie, po co są i do czego służą koprocesory? Mam nadzieję, że chociaż większość z Was nieprędko będzie miała okazję (i potrzebę!) korzystania z komputera wyposażonego w koprocesor, zawarte w tym atrykule wiadomości przydadzą się Wam — dla własnej satysfakcji.

Marcin Borkowski

INSTRUMENTY Z TEKSASU

Wraz ze wzrostem możliwości obliczeniowych kalkulatorów posługiwanie się nimi stawało się coraz trudniejsze. Konieczność tworzenia i zapisywania długich algorytmów obliczeniowych służących do rozwiązywania złożonych problemów zrodziła potrzebę wyposażenia kalkulatora w mechanizm spotykany do tej pory tylko w maszynach cyfrowych — pamięć programu.

Dzięki niej użytkownik mógł zapisywać potrzebne mu procedury obliczeniowe bezpośrednio w kalkulatorze, a ten mógł korzystając z wprowadzonego programu dokonywać wielokrotnie tych samych obliczeń dla różnych danych wejściowych pozwalając np. na stosowanie metody kolejnych przybliżeń bardzo przydatnej przy obliczeniach konstrukcyjnych. Z zastosowniem pamięci programu wiąże się również formalizacja zapisu typowych rozwiązań problemów obliczeniowych dla wielu dziedzin nauki i techniki, można je wykorzystywać bez konieczności opracowywania własnego programu.

Systematyczne przedstawienie możliwości kalkulatorów programowanych jest wobec braku standardu zadaniem dość trudnym dlatego postanowiłem posłużyć się opisem rodziny produkowanej przez firmę Texas Instruments (SR56, TI58, TI58C, TI59). Dzięki relatywnie niskiej cenie są to kalkulatory dość popularne w naszym kraju, a jednocześnie ich duże możliwości pozwalają na w miarę pełne zapoznanie z zagadnieniami programowania.

Przedstawione kalkulatory można podzielić na dwie grupy, model SR56 posiada średnią moc obliczeniową oraz minimalną liczbę funkcji wspomagających programowanie, jest przykładem popularnego kalkulatora programowanego. Pozostałe trzy modele dysponują dużą mocą obliczeniową i wspólnym zbiorem instrukcji. Różnice między nimi istnieją w wielkości i rodzaju pamięci oraz w posiadanych urządzeniach peryferyjnych, można je zaliczyć do grupy półprofesjonalnej.

Na początek kilka słów o możliwościach arytmetycznych kalkulatorów programowanych. Posiadają one typowy zestaw operacji i funkcji matematycznych spotykany w kalkulatorach naukowych, system algebraiczny z preferencją działań i wielopoziomowy układ nawiasów. Dostępnych jest kilka sposobów zapisu liczb (zmiennoprzecinkowy, stałoprzecinkowy, naukowy i inżynierski), a obliczenia wykonywane są z dokładnością do 13 miejsc znaczących. Znacznie rozbudowany w porównaniu z kalkulatorami inżynierskimi jest sposób komunikowania się z pamięcią danych. Zwiększona jest ilość rejestrów pamięci dostępnych dla użytkownika, SR56 posiada ich 10, zaś w przypadku pozostałych modeli ilość ta jest zmienna, zależy od aktualnego podziału pamięci operacyjnej pomiędzy dane a program i może wynosić od 0 do 100 dla TI59 lub od 0 d 60 dla TI58 i TI58C. Każdy rejestr pamięci posiada numer i za jego pomocą jest identyfikowany przy operacjach zapisu, odczytu i zamiany z rejestrem X.

Przypomnę, że rejestr X jest jednym z rejestrów wewnętrznych kalkulatora o roli podobnej do akumulatora w mikroprocesorze, do tego rejestru wprowadzane są liczby z klawiatury i jego zawartość widnieje na wyświetlaczu. Oprócz typowych operacji przenoszenia danych, na rejestrach pamięci wykonywać można operacje arytmetyczne, których pierwszy argument a następnie wynik znajdują się w wybranym rejestrze, a drugim argumentem jest aktualna zawartość X. Słabszą stroną prezentowanych kalkulatorów jest sposób przedstawiania zapisu programu, w związku z tym, że dysponują one typowym wyświetlaczem siedmiosegmentowym wszystkie instrukcje programu reprezentowane są kodowo. Utrudnia to zapisywanie i modyfikacje programów w kalkulatorze. Zasada tworzenia kodu dla danej instrukcji oparta jest na ogół o współrzędne danego klawisza na klawiaturze. Dla ułatwienia posługiwania się tym zapisem w modelach z symbolem TI stosuje się specjalne nakładki na klawiaturę na których każdej funkcji przyporządkowany jest odpowiedni kod. Jedynym ułatwieniem przy posługiwaniu się tym systemem zapisu jest samoczynne tworzenie kodów podczas wpisywania programu z klawiatury co upraszcza przenoszenie programu z papieru do pamięci kalkulatora.

Jednym z podstawowych pojęć związanych z programami kalkulatorowymi jest krok, jednostka długości programu odpowiadająca pojedyńczemu użyciu funkcji, zapisaniu numeru rejestru pamięci lub etykiety. Liczba kroków programu jaką można zapisać w kalkulatorze jest jednym z parametrów mocy obliczeniowej, przykładowo dla SR56 wynosi ona 100 dla TI58 i TI58C od 0 do 480, a dla TI59 od 160 do 960. Zmienna pojemność pamięci programu związana jest z elastyczną strukturą umożliwiającą zmianę proporcji pomiędzy pamięcią danych a pamięcią programu, za pomocą bloków których zastosowanie można zmieniać, każdemu z nich odpowiada 10 rejestrów pamięci lub 80 kroków programu.

Aby zapisany w pamięci program można było wykonać w określony sposób każdy krok opatrzony jest etykietą numeryczną odpowiadającą położeniu kolejnych instrukcji, umożliwia ona wykonywanie wewnątrz programu skoków jak również ułatwia odszukanie wybranych jego fragmentów celem dokonania zmian i poprawek.

Po zapoznaniu się z podstawowymi pojęciami możemy przystąpić do pisania programu, w tym celu należy postępować podobnie jak podczas zapisu operacji na zwykłym kalkulatorze z tą różnicą, że w miejscach gdzie następuje wprowadzanie danych i wyprowadzanie wyników należy posłużyć się instrukcjami zapisu i odczytu odpowiednich rejestrów pamięci dokonując na ich zawartości operacji podobnie jak na zmiennych. Kiedy już mamy gotowy zapis programu należy przestawić kalkulator w opcję programowania i wprowadzić go do pamięci posługując się klawiaturą podobnie jak podczas normalnych obliczeń. Po wpisaniu całego programu należy powrócić do normalnego trybu pracy i rozpocząć testowanie jego poprawności, w związku z tym, że program musi być testowany po każdym wprowadzeniu do pamięci warto posiadać zestaw danych kontrolnych wraz ze wszystkimi wynikami pośrednimi który ułatwi szybkie odszukanie błędów. Podczas programownia można przeglądać tekst programu zarówno do tyłu jak i wprzód. W kalkulatorach TI można również rozsuwać tekst programu dla wprowadzenia np. przeoczonej instrukcji lub też przesuwać cały program kasując zbędne kroki programu. Brak w SR56 tych dwóch instrukcji jest ogromnym utrudnieniem przy poprawianiu błędów, zmuszającym do wielokrotnego wpisywania programu w przypadku pomyłek.

Porównując system programowania kalkulatorów z typowymi językami wyższego rzędu odnaleźć można wiele podobieństwa, szczególnie w zakresie instrukcji związanych ze strukturą programu. Możliwe jest wykonywanie skoków warunkowych i bezwarunkowych, można stosować podprogramy, istnieje również instrukcja cyklu podobna w działaniu do pętli FOR-NEXT. Znaczne różnice dotyczą jednak systemu zapisu tych działań w programie np. skoki warunkowe są związane w kalkulatorach Texas'a z istnieniem dodatkowego rejestru pamięci p nazwie t, w momencie natrafienia przez zrealizowany program na instrukcję skoku warunkowego wartości rejestrów t i X jest porównywana i jeśli warunek logiczny jest spełniony, to program skacze do miejsca, którego etykieta zapisana jest bezpośrednio po instrukcji warunkowej, a jeśli nie jest spełniony, to program jest wykonywany dalej z pominięciem zapisanej etykiety. Z działaniem podprogramów związane jest ograniczenie tzw. zanurzenia kolejno wywołujących się procedur, w SR56 istnieje system czterokrotnego wywołania, a w pozostałych modelach sześciokrotnego.

Przedstawione powyżej instrukcje stanowią podstawowy zestaw spotykany w kalkulatorach programowanych o możliwościach zbliżonych do SR56, wprowadzenie nowych modeli o większej mocy obliczeniowej wiązało się nie tylko ze zwiększeniem objętości pamięci ale też z opracowaniem nowego zestawu instrukcji który lepiej wspomagał by programowanie jak również z uniezależnieniem pamięci od źródeł zasilania, stosowana dotychczas powodowała utratę programu i danych po wyłączeniu kalkulatora, a więc konieczność ręcznego wprowadzania programu przed każdym jego użyciem.

Dalszy opis dotyczyć będzie następców SR56: modeli TI58, TI58C i TI59, w których zastosowano szereg rozwiązań ułatwiających posługiwanie się zwiększoną pamięcią. Jednym z najciekawszych jest dołączenie do kalkulatora modułu zewnętrznej pamięci stałej, która zawiera programy przygotowane przez producenta, można je wykorzystywać albo oddzielnie jako dodatkowe funkcje kalkulatora, lub też stosować jako segmenty we własnych programach, można również po przepisaniu do pamięci wewnętrznej dokonywać w nich dowolnych modyfikacji w celu dostosowania do własnych potrzeb. Wraz z kalkulatorem dostarczany jest moduł z programami matematycznymi, obejmuje on rachunek macierzowy obliczenia na liczbach zespolonych, obliczenia statystyczne, typowe metody numeryczne, konwersje jednostek miar i wag oraz wiele innych zagadnień. Cały moduł zawiera 25 wielosegmentowych programów o łącznej długości 5000 kroków.

Dużym ułatwieniem jest wprowdzony nowy sposób adresowania pamięci programu, oprócz stosowanych do tej pory etykiet numerycznych, będących numerem kolejnym kroku, do zaznaczania miejsc w programie można używać etykiet symbolicznych — w tym celu należy posłużyć się instrukcją etykiety, po której umieszcza się dowolną inną instrukcję, cyfrę, lub znak np. literę znajdującą się na klawiaturze. Powoduje to oznaczenie danego punktu programu użytym symbolem, który traci w tym miejscu swe zwykłe znaczenie. Po zastosowaniu tego typu oznaczeń nie trzeba już podawać w instrukcjach wymagających etykiety jej postaci numerycznej, wystarczy podać zastosowany symbol, a kalkulator odszuka go w pamięci i wykona skok w oznaczone miejsce. Jeśli użytym symbolem jest litera, to uzyskujemy jeszcze jedną możliwość korzystania z programu lub jego części. Klawisz z wizerunkiem tej litery staje się klawiszem funkcyjnym-naciskając go powodujemy automatyczne uruchomienie oznaczonego fragmentu programu. Daje to ogromne możliwości rozszerzenia standardowego zestawu funkcji oraz możliwość posługiwania się kilkoma programami znajdującymi się równocześnie w pamięci kalkulatora. Jednak największą zaletą etykiet symbolicznych jest ułatwienie zmian i modyfikacji programu. Dotychczas każde wprowadzenie nowych instrukcji pomiędzy istniejące wymagało ręcznej zmiany wszystkich etykiet używanych w programie, nowy system adresowania automatycznie kompensuje wszelkie zmiany. Niestety oprócz zalet posiada on także wady. Podstawową jest niejednoznaczność zapisu instrukcji w programie, określanie czy instrukcja użyta w danych miejscu oznacza samą siebie czy jest tylko symbolem etykiety wymaga analizy tekstu programu i przyczynia się do zmniejszenia przejrzystości jego zapisu.

Stosowany w SR56 sposób wyznaczania argumentów dla instrukcji wymagał, aby numery rejestrów pamięci lub etykiety numeryczne zapisane były jawnie w tekście programu. W nowych modelach kalkulatorów można w miejscu argumentu posłużyć się instrukcją adresowania pośredniego wraz z numerem rejestru pamięci, w którym znajduje się argument. Ten trochę zawiły sposób umożliwia uzyskanie nowych zastosowań dla wielu instrukcji, możliwe jest np. wykonywanie skoku liczonego, w którym adres docelowy wyznaczany jest za pomocą funkcji arytmetycznej oraz cykliczne wywoływanie kolejnych rejestrów pamięci przydatne przy obliczeniach macierzowych.

Zwiększony rozmiar pamięci programu umożliwił pisanie programów modułowych o zmiennej strukturze dostosowywanej każdorazowo do rozwiązywanego problemu, ułatwieniem dla tego rodzaju zmian jest sterowana instrukcja skoku, za jej pomocą można przed uruchomieniem programu lub podczas jego pracy dokonywać zmian w połączeniu poszczególnych segmentów. Działanie jej jest podobne do instrukcji skoku warunkowego, nie następuje jednak kontrola warunku logicznego a sprawdzany jest stan jednego ze specjalnych znaczników, które mogą być ustawiane za pomocą oddzielnej instrukcji (dostępnych jest 10 takich znaczników).

Oprócz przedstawionych zmian o charakterze zasadniczym w nowych modelach firma Texas Instruments wprowadziła szereg drobnych udoskonaleń dotyczących m.in. rozszerzenia standardowego programu statystycznego o drugą zmienną, programowej reakcji na błędy obliczeń, nowych działań na rejestrach pamięci i informacji o aktualnym podziale pamięci.

Po przedstawieniu zmian dotyczących możliwości obliczeniowych kilka słów na temat zmian w budowie nowej rodziny kalkulatorów. Oprócz wspomnianego rozszerzenia pamięci w TI58C zastosowano bateryjne podtrzymanie jej zawartości umożliwiające przechowywanie programów i danych TI59 w budowany został czytnik kart magnetycznych umożliwiający przechowywanie poza kalkulatorem programów i danych w postaci umożliwiającej szybkie użycie. Na jednej karcie można zapisać dwustronne 480 kroków programu lub odpowiednią ilość rejestrów pamięci. Dla zabezpieczenia praw autorskich zapisany program można zabezpieczyć przed wylistowaniem. Za pomocą kart magnetycznych zestawiać można własne biblioteki programów lub korzystać z bibliotek przygotowanych przez firmę.

Aby ujednolicić wykorzystanie klawiszy funkcyjnych, w obudowie kalkulatora wykonano odpowiednie wycięcie, w którym umieszczać można karty magnetyczne, na odwrocie których zapisuje się znaczenie przypisywane poszczególnym klawiszom, i podobnego formatu karty z opisem funkcji programów znajdujących się w module pamięci. Te drobne z pozoru udoskonalenie znacznie ułatwia posługiwanie się predefiniowalną klawiaturą.

W tym z konieczności skrótowym opisie starałem się przedstawić ogromne możliwości kalkulatorów programowanych, które w połączeniu z ich relatywnie niską ceną pozwalają mi konkurować przy rozwiązywaniu problemów obliczeniowych z komputerami domowymi szczególnie w przypadku pracy poza domem, w szkole lub w podróży.

Adam Zakrzewski



STRUKTURY DANYCH (V)

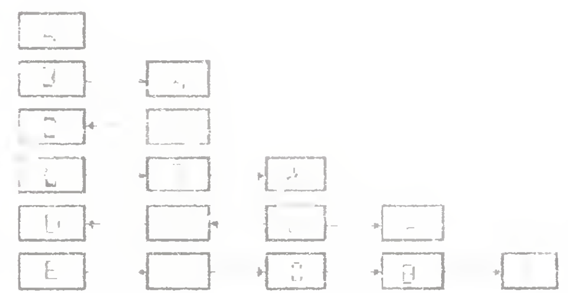
Dziś pragnę zaprezentować stos, czyli koncepcję dostępu do danych pozwalającą w prosty sposób rozwiązywać kilka ważnych problemów związanych z działaniem komputerów. Zanim jednak to zrobię musimy rozprawić się ostatecznie z listami.

Niejako z przeniesienia został nam problem gospodarki wolnym miejscem w pamięci, a konkretnie właściwego wykorzystania fragmentów zwalnianych przy usuwaniu elementów z listy. Takie wolne bloki pojawiają się dynamicznie, w różnych miejscach pamięci. Aby poradzić sobie z danymi, które pojawiały się dynamicznie zastosowaliśmy listy. Aby poradzić sobie z fragmentami wolnej pamięci zróbmy to samo. Skoro jest to wolne miejsce więc nie ma w nim żadnych danych, a jeśli tak to nikomu nie będzie przeszkadzać że na tym miejscu napiszemy sobie adres początku następnego wolnego bloku. W ten sposób stworzymy listę wolnych obszarów, z której będziemy mogli pobierać elementy gdy zajdzie potrzeba znalezienia miejsca na zapisanie nowych danych.

Warto też powiedzieć o jeszcze jednym, generalnym zastosowaniu list. Skoro ułatwiają one utrzymanie w porządku obiektów, które są często dodawane i usuwane (czyli pobierane), to powinny nam umożliwić realizowanie postulatu: obiekty należy pobierać w takiej kolejności w jakiej były wpisywane. Inaczej mówiąc ten, który został wpisany jako pierwszy powinien być wyznaczony do pobrania również jako pierwszy. Po angielsku ta strategia jest określana krótko First-In-First-Out, lub w skrócie FIFO. Podaję te terminy, gdyż można je dość często spotkać w literaturze fachowej. Po polsku na taką strukturę mówimy krótko kolejka i jest to chyby zupełnie naturalna nazwa — w prawdziwej kolejce ludzie są obsługiwani w takiej kolejności w jakiej przybywają do sklepu.

Postawione przez nas zadanie można zrealizować na wiele sposobów, jednak przechowując dane o kolejkowanych obiektach w tablicy do której mamy tylko zwykły bezpośredni dostęp będziemy mieli sporo problemów za zapamiętywaniem kolejności. Korzystając z list mamy bardzo proste rozwiązanie: nowe obiekty dopisujemy zawsze na początku list (tzn. staje się on pierwszym na liście), zaś pobieramy zawsze ostatni.

Popatrzmy jak budowana jest lista, jeśli przybywały elementy w następującej kolejności: A, B, C, D, E



Oraz w jakiej kolejności będziemy zdejmować elementy, biorąc zawsze ostatni:

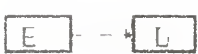
pobieramy A, pozostaje lista:



pobieramy B, pozostaje lista:



pobieramy C, pozostaje:



itd. Jak widać dostajemy elementy rzeczywiście w kolejności wpisywania. Zwykła lista z określonymi powyżej zasadami dopisywania i pobierania elementów tworzy kolejkę. Niestety jednak, mimo prostoty, nie jest to metoda bardzo efektywna. Przecież aby móc pobrać ostatni element musimy zawsze przejść przez całą listę (mówiliśmy o tym w zeszłym miesiącu). Tak więc operacja pobrania jest dość kosztowna. Moglibyśmy zmniejszyć te koszty mając dodatkową strzałkę pokazującą nam zawsze aktualny koniec listy:



Ponieważ jednak po pobraniu elementu A strzałka ta staje się zupełnie [illegible]



Dzięki temu [illegible] zawsze usuwając ostatni element bez trudności znajdujemy ten, który teraz będzie ostatni — pokazuje go strzałka skierowana w lewo.

Tak naprawdę, to zyskaliśmy nawet więcej, po naszej liście można poruszać się teraz w dwóch kierunkach. Tak też nazywa się ta struktura: lista dwukierunkowa. Oczywiście ma ona też i minusy, wymaga więcej miejsca w pamięci do zapisania takiej samej liczby rekordów, gdyż na każdy z nich potrzebujemy teraz dwóch wskaźników.

Opisana metoda implementacji kolejki za pomocą listy jednokierunkowej (dopisywać nowe na początku, pobierać z końca), okazała się być bardzo pożyteczna — dzięki niej poznaliśmy nową, pożyteczną strukturę list dwukierunkową. Ale tak naprawdę to można zrealizować kolejkę w efektywny sposób w oparciu o listę jednokierunkową. Jak — to zadanie dla Was a rozwiązanie podam za miesiąc.

Tak czy inaczej kolejkę zrealizowaliśmy pomyślnie i na razie zostawimy listy w spokoju, za to przyjrzyjmy się innej sytuacji z życia wziętej. Na ślepy boczny tor kolejowy wtaczamy [illegible] oznaczmy je literami A, B, C, D, E



W jakiej kolejności możemy wyciągać je z powrotem. O ile są to zwykłe, nie mające skrzydeł wagony, to jedyną możliwością jest E, D, C, B, A, czyli kolejność dokładnie odwrotna do tej, w której były wtaczane! Element, który przybył ostatni wychodzi pierwszy, nasza bocznica realizuje więc zasadę: Last-In-First-Out w skrócie LIFO. W polskiej terminologii, na strukturę pamięci o tak zdefiniowanym dostępie do danych mówimy stos. Ze stosem możemy wykonywać dwie operacje: wkładania elementu na stos, oraz pobierania wierzchołka stosu (angielskie nazwy tych operacji to odpowiednio PUSH i POP). Każdy dokładany element ląduje po prostu na wierzchołku, każde pobranie zdejmuje jeden element z wierzchołka (odsłaniając tym samym do pobrania następny). Dodatkowo, potrzebna jest możliwość sprawdzenia czy stos jest pusty (z pustego stosu nie można nic pobrać).

Niektóre komputery mają stos zrealizowany w sposób sprzętowy. Oznacza to, że część pamięci operacyjnej jest zarezerwowana na obszar stosu systemowego, a w repertuarze instrukcji procesora znajdują się wkładanie porcji danych (np. jednej komórki pamięci) na stos oraz pobieranie danych ze stosu. Do realizacji tego zadania wcale nie są potrzebne niezwykłe układy, wystarczy zwykła pamięć o dostępie bezpośrednim. Przekonamy się o tym realizując strukturę stosową w tablicy jednowymiarowej ST[1 1000], która jest [illegible]

[illegible] jest pusty, to wartość TOP [illegible] wpisaniu jej do elementu wskazywa-

KOPERTA

wkładaj delikatnie

nie zginaj

nie dotykaj

uwaga na magnesy

10° – 52° C

zabezpiecz po użyciu

Pegaz Ass

Commodore

Bajtek

Wszyscy wiemy jak łatwo niszczą się koperty do dyskietek 5,25 cala. Wykonane zazwyczaj z cienkiego papieru ulegają rozdarciom, naderwaniom, zalaniom herbatą, pogryzieniom przez psy i innym podobnym wypadkom. Co prawda kosztowne dyskietki takich firm jak np. MAXELL, BASF, czy 3M znajdują się w kopertach wykonanych ze specjalnego kartonu, lecz nieczęsto przecież zdarza nam się posiadanie większej liczby takich dyskietek.

Idąc z pomocą posiadaczom dużych zbiorów opracowany został szablon koperty do standardowej dyskietki 5,25". Wystarczy zrobić jej KSEROKOPIĘ, wyciąć i skleić. Wskazane jest użycie kleju, który pozostaje elastyczny po zaschnięciu. Kleje roślinne i tzw. biurowe mają skłonności do wykruszania się. Przedostanie się takiej zaschniętej grudki kleju na powierzchnię dyskietki może łatwo doprowadzić do jej trwałego uszkodzenia, lub co gorsza do zniszczenia głowicy w stacji dysków.

Robiąc kopie proszę nie zapomnieć i koniecznie dowiedzieć się czy dany kserograf robi dobitki 1:1, niektóre starsze modele często zmieniają format kopii po kilka milimetrów w każdą stronę, a co za tym idzie nasza dyskietka może „latać" w kopercie lub się nie zmieści.

Wolna powierzchnia na kopercie przygotowana została jako miejsce na wpisanie tytułów kolejnych zbiorów na dysku.

Prezentowana dziś koperta przygotowana została z myślą o użytkownikach COMMODORE. Już niedługo opublikujemy koperty dla posiadaczy ATARI i IBM, oczywiście ze specjalną szatą graficzną.

Bajtek

KOPERTA

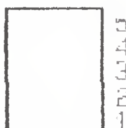
---

nego przez TOP, pobranie zwraca liczbę z elementu wskazywanego przez TOP i zmniejsza wartość TOP o 1.

Wykonanie PUSH(A) wymaga więc operacji:
TOP=TOP+1
ST[TOP]=A
zaś POP
POP=ST[TOP]
TOP=TOP-1

Wykonajmy kilka przykładowych operacji, poczynając od pustego stosu. Aby ułatwić odszukanie miejsca pokazywanego przez TOP, z prawej strony tablicy podaję numery jej elementów:

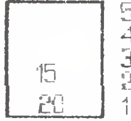
TOP = 0



PUSH (20)
TOP = 1



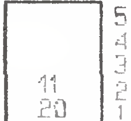
PUSH (15)
TOP = 2



POP – zwraca 15
TOP = 1



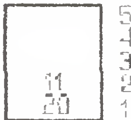
PUSH (11)
TOP = 1



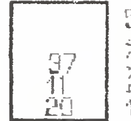
PUSH (65)
TOP = 3



POP zwraca 65
TOP = 2



PUSH (37)
TOP = 3



Oczywiście, tak jak w przypadku realizacji list w oparciu o tablice, jest możliwość pobrania lub wpisania wartości w dowolnym miejscu tej tablicy, tyle że także dla stosu taka operacja jest bezsensowna. Jeśli nie skorzystamy z wartości zmiennej TOP, to pobierając zawartość komórki nie możemy nawet mieć pewności czy w tej komórce cokolwiek było zapisane. Tak więc, jedyną możliwość czytania danych, jest pobieranie ich w kolejności narzuconej przez strukturę stosową.

Po co to wszystko? Trudno zaprzeczyć, że opisane powyżej konstrukcje wydają się dość nienaturalne i raczej bezużyteczne. Zaś szukając przykładów stosu nie w informatyce lecz w praktyce łatwo znajdziemy miejsca, w których należy... unikać tej struktury. Jeśli np. stos zbudowany będzie nie z liczb, lecz z puszek konserw w magazynie, to zawsze będziemy zużywali pobrane z wierzchołka naświeższe, natomiast te leżące na dnie mają spore szanse się popsuć. Tak więc towary w magazynie powinny tworzyć kolejkę a nie stos.

Ponieważ znalezienie dobrych zastosowań dla stosu może być rzeczywiście problemem więc nie przejmujcie się jeśli nic Wam nie przyjdzie do głowy przez najbliższy miesiąc. Zapewniam Was jednak, że takie zastosowania naprawdę istnieją i omówimy je w następnym numerze.

Andrzej Krull

ATARI
ZX SPECTRUM
INSTRUKCJE ● OPISY LITERATURA

Szkoły i Kluby-Zniżka
Katalogi-Gratis
Co piąty program-Gratis
Wysyłka na cały kraj
Wypożyczalnia programów
D.H. „SEZAM" IIp.g. 16.00—19.00
00-849 Warszawa UPT 66, skr.p. 14

D-122

SERWIS KOMPUTERÓW
TEST

40-164 Katowice ul. Modrzewiowa 24/33
poleca naprawy:
- ATARI 600, 800, 65, 130 XL,XE
- COMMODORE 16, 116, +4, 64, 128
- DISC DRIVE 1541, 1570, 1571. 1050
- MAGNETOFONY COMMODORE
- DRUKARKI

godz. 9—11, 16—18

SB 30

TOMBAT
XL ATARI XE

- gry i programy użytkowe
- co piąty program gratis
- inne bonifikaty
- opisy gier i instrukcje
- pomoc dla początkujących
- wysyłka na cały kraj
- katalog gratis

WYPOŻYCZALNIA TOMBAT XL ATARI XE

Nasz adres:
ul. Magistracka 27 m 26
01-413 Warszawa
Tel. 363-078 godz. 12-20
Zapraszamy!

SB-12

Studio Komputerowe
POMAREX
poleca: programy, literaturę i pośrednictwo w obrocie sprzętem komputerowym ATARI, AMSTRAD, SPEKTRUM
Lublin ul. Buczka 25

Sd 307

Rewelacyjnie niskie ceny!!! Programy dla Atari XL/XE. Informacje (koperta + znaczek) Alek i Robert Corp, ul. Zielona 14 lub ul. Niecała 9, 36-100 Kolbuszowa.

(Sd 303)

COMPUTER SERVICE

MS elektronik
naprawy komputerów:
Spectrum 48k, +, 128, +2, +3
Amstrad- Schneider
Sharp
Drukarki, Interfejsy
Wyjścia monitorowe
Czynne: od 9.00—16.00
MSelektronik Legionowa 23, 00-343 Warszawa
Dojazd: 105, 305, F (jelonki)
tel. 37-76-65.

(K-118)

AMIGA COMMODORE CLUB

Jeżeli jesteś szczęśliwym posiadaczem AMIGI, nie zamierzasz sprzedawać oprogramowania, a przy tym pragniesz nawiązać kontakt z innymi użytkownikami tego komputera, przyłącz się do najliczniejszego klubu AMIGI w Polsce. Członkostwo w ACC jest bezpłatne, a umożliwia Ci dostęp do ogromnej ilości oprogramowania i literatury (m.in. odpłatnego miesięcznika klubowego). Twojego zgłoszenia listownego oczekujemy pod adresem: Marek Hyla, osiedle Kolorowe 9/16, 31-939 Kraków, ACC. Telefon 44-43-68.

(SB-60)

STUDIO KOMPUTEROWE
HIT – BIT

programy, literatura, osprzęt komputerowy
ATARI COMMODORE SPECTRUM AMSTRAD
Warszawa: D.H. „Merkury"
przy Pl. Komuny Paryskiej,
tel. 39-92-51 w. 69
Wysyłka: HIT — BIT
00-967 Warszawa 86
skr. poczt. 168
katalog po załączeniu koperty.

D-51

ZX SPECTRUM, TIMEX, ATARI
- **programy użytkowe, edukacyjne, gry**
- **programy dla rzemiosła (receptury, kalkulacje, remanenty)**
- **instrukcje do programów**
- **informacje po nadesłaniu koperty zwrotnej ze znaczkiem**
- **wysyłka na cały kraj — rachunki**

„P.K.T.S." Studio Komputerowe
00-103 Warszawa
ul. Krolewska 43 m 25

(SB-59)

COMMODORE 16,116.4/PLUS ATARI XE. XL. ST SHARP Studio komputerowe „CANON" Proponuje bogatą ofertę oprogramowania Koperta zwrotna Chorzów 41-506 ul. Karłowicza 23/12

D-87

Naprawa komputerów ATARI, COMMODORE, IBM, SPECTRUM oraz urządzen peryferyjnych Warszawa tel. 22–07–85

D-84

MÓZGOPROCESOR! — rewelacyjną polską grę przygodową dla ZX Spectrum (program + kaseta + opis 6100 zł) otrzymasz pisząc: COMPUTER ADVENTURE STUDIO, Bochnia 32-700, ul. Kazimierza Wielkiego 37/45.

(SB 20)

Wymienię programy dla SPECTRUM, ATARI, COMMODORE. Janusz Wałaszek, skrytka pocztowa 1 33-106 Tarnów 8.

(SB 25)

CZĘŚCI ELEKTRONICZNE

pochodzenia zagranicznego za złotówki układy TTL, LS, CMOS, stabilizatory, mikroprocesory, pamięci.
Krótkie terminy dostaw, atrakcyjne ceny.
Pośredniczymy także przy sprzedaży.
„MARITEX" tel. 22-02-89
Gdynia, ul. Batalionów Chłopskich 3
tlx 054622

K-190

IBM XT,AT/386 COMMODORE ATARI SPECTRUM (TIMEX)

Sprzęt, programy, opisy
ATUT Sp. z o.o.
ul. Weteranów 38
20–045 Lublin

K-133

OTV RADZIECKIE
JUNOST ELEKTRONIKA SILELIS
oraz STACJONARE:
NAPRAWA ● KINESKOPY ● DEKODERY PAL ● FONIA CCIR
WEJŚCIA MONITOROWE
INTER-SERWIS,
ul. Rutkowskiego 10/12
00–020 Warszawa, tel. 27–47–72

D-76

NAJTAŃSZE (sprawdź) programy kasetowe ATARI XL/XE **i NAJKORZYSTNIEJSZY** (koniecznie sprawdź!) sposób realizacji zamówień. Bezpłatne informacje: **UNICO**, skr. 39, 41-800 ZABRZE.

G-63

Największy wybór oprogramowania do ATARI ST, XE, XL, COMMODORE 64, AMIGA, na najkorzystniejszych warunkach udostępnia HOOZY 81-706 SOPOT BOH.MONTE CASSINO 21 D.
Prosimy o kopertę i znaczek

G-127

SPECTRUM

najtańsze oprogramowanie
Nowości!
Komuny Paryskiej 36a/16
Szczecin

G-46

ZX Spectrum
Ekspresowe naprawy klawiatur
NiKUE ul. Meissnera 14 m 1,
03–982 Warszawa tel. 15–93–38
wieczorem

D-73

MIKROSERVICE COMMODORE — 64/128, AMIGA, ATARI, SPECTRUM, IBM — PC/XT/AT INTERFEJSY — CENTRONICS, RS 232, DIGITIZER, FINAL, SPEEDDOS, PROGRAMATORY.
EUROKARTY - Z80, 6502, 68000 A/C CP/M
01-911 WARSZAWA, ANDERSENA 3/103.

D-52

KOMPUTEROWA INTELIGENCJA



Wielokrotnie pisaliśmy na łamach „Bajtka" o potrzebie upowszechniania i popularyzowania osiągnięć nauki i techniki. Nie jest to łatwe, gdyż wymaga zarówno specyficznych umiejętności dziennikarskich, jak i odrobiny talentu. Dlatego tak niewielu mamy w Polsce dobrych popularyzatorów nauki i techniki. Ale dlatego również należy się tej umiejętności uczyć — i to z im wcześniejszych lat się zacznie, tym lepiej.

Poniższy tekst przyniósł nam przed wakacjami do redakcji Piotruś Kos, uczeń VI klasy w jednej ze szkół podstawowych w warszawskim osiedlu Bródno. Drukujemy go — świadomi pewnych ułomności warsztatowych, których nie sposób uniknąć, gdy ma się 13 lat i dopiero debiutuje w dziennikarsko-komputerowej redakcyjnej rodzinie.

(ws)

CZY KOMPUTER MOŻE BYĆ NIEZASTĄPIONY?

Wielcy naukowcy, wybitni informatycy i całe sztaby programistów pracują nad tym, aby na to pytanie móc odpowiedzieć twierdząco. Bez wątpienia byłby to największy sukces w dziejach informatyki.

Wbrew pozorom wcale nie jest to pytanie proste, choć na pierwszy rzut oka może wydać się co najmniej infaltylne, a w gruncie rzeczy niezwykle trudno byłoby dać na nie jednoznaczną odpowiedz.

Dynamiczny rozwój wszystkich dziedzin życia pociąga za sobą szybki wzrost zastosowań komputera, a co za tym idzie, coraz większą wszechstronność i niezawodność systemów komputerowych. Panuje obecnie zupełnie zresztą słuszny pogląd, że komputery są dla ludzi, a nie odwrotnie. W związku z tym podąża się w kierunku jak największej miniaturyzacji oraz prostoty w obsłudze systemu. Mimo to powstające masowo oprogramowanie użytkowe ma nierzadko kilkusetstronicowe instrukcje obsługi.

Chociaż oprogramowanie jest bez wątpienia najważniejsze przy użytkowaniu komputera potrzebny jest również sprzęt, na którym mogłoby ono działać. Jest on coraz doskonalszy, lecz komputer idealny nie istnieje. Następnym krokiem w kierunku jego stworzenia jest NeXT (dosłownie: następny), najnowszy produkt firmy NeXT Company, której właścicielem jest Steven Jobs

Komputer ten został określony jako „o klasę lepszy od wszystkiego, co dotychczas oglądano" Oparty na trzech procesorach Motorola: głównym 68030; procesorem 68882 do obliczeń numerycznych i 56001 do syntezy dźwięku i do grafiki, posiada pamięć RAM o pojemności 8MB, dysk optyczny Canon 256 (!)MB oraz rozdzielność ekranu 832 na 1120 punktów w czterech odcieniach szarości. Wyposażono go również w system operacyjny UNIX.

NeXT jest jednak tylko narzędziem, które staje się bezużyteczne bez odpowiedniego oprogramowania. Lecz kiedy takie powstanie, komputer ten, jak wiele innych, zostanie wdrożony do wykonywania wielu pożytecznych prac, a zajęć dla niego z pewnością nie zabraknie. Nie ma bowiem branży, w której nie miałyby udziału komputery.

Spróbujmy wyliczyć główne zastosowania tych skomplikowanych urządzeń.

Komputery magazynują dane, przetwarzają teksty, wykonują astronomiczne obliczenia matematyczne, w szpitalach potrafią kontrolować pracę serca chorego, w wielkich ośrodkach naukowych weryfikować badania i sprawdzać pomiary. W fabryce nadzór nad produkcją taśmową najlepiej powierzyć szesnatobitowemu systemowi, a zarządzanie przedsiębiorstwem komputerowi IBM, nie mówiąc już o tym, co jest wykorzystywane jako „mózg" bazy wojskowej wyposażonej w broń atomową. Nie sposób wyliczyć tutaj wszystkich zastosowań, gdyż praktycznie każdy program niesie za sobą nowe możliwości wykorzystania komputera.

Nie znaczy to wcale, że komputer może wszystko, bo na przykład

KOMPUTER MISTRZEM SZACHOWYM...

nigdy nie zostanie!

Arcymistrz dr Helmut Pfleger powiedział — „Szachy to zbyt skomplikowana gra; już jej pierwsze dwadzieścia dwa ruchy można rozegrać na więcej sposobów, niż jest atomów we Wszechświecie" Nie przeszkodziło mu to być współtwórcą trzydziestodwubitowego komputera szachowego Mephisto.

Widzimy więc już, że komputer nigdy w szachy dobrze grać nie będzie, choć są programy pozwalające rozgrywać partie na ekranie, np. Cheesmaster.

Skoro nie nieograniczone, to wobec tego

JAKIE SĄ MOŻLIWOŚCI KOMPUTERA?

Popatrzmy na to zagadnienie z technicznego punktu widzenia. Jedyną możliwością komputera jest wykonywanie skomplikowanych obliczeń matematycznych ze skończoną dokładnością. Inna sprawa, że na tych obliczeniach opiera się bogaty system o dużych możliwościach.

Mimo to inteligentny komputer pozostaje nadal marzeniem. Jak dotąd symulacja ludzkiego myślenia jest niemożliwa. Nawet najgenialniejsze Superkomputery przez duże S nie są w stanie dokonać wyboru lub trafnie przeprowadzić analizy nieznanej im sytuacji. Dysponują jedynie ograniczoną ilością struktur pseudologicznych.

Chciałbym tu uspokoić fanatyków sztucznej inteligencji i tych, którzy chcieliby widzieć swoje ośmiobitowe Atari w roli filozofa. Komputer nadal pozostaje jednym z najinteligentniejszych urządzeń świata.

Po co więc było to wszystko?

Chodziło o udowodnienie, że komputer choć ważny i potrzebny, nie jest wszystkim. Bo tak naprawdę, najważniejszy, i jak dotąd niezastąpiony, jest człowiek.

Nie oszukujmy się — w porównaniu z człowiekiem najcudowniejszy nawet komputer jest po prostu głupi

Marcin Kos
lat 13

ABC Data







Drodzy Przyjaciele

Mam na imię Marta. W firmie ABC DATA zajmuję się prowadzeniem korespondencji. Otrzymuję z Polski wiele listów, często ozdobionych grafiką komputerową i zabawnymi rysunkami. Wiele z nich jest tak pięknych, że myślę o zrobieniu wystawy. Na wszystkie listy odpowiadam, choć nie jest to łatwe. Mam wiele obowiązków, bo tak jak wy chodzę do szkoły i czasu mam niewiele. Dlatego wybaczcie mi jeżeli na odpowiedź będziecie musieli poczekać. Ale na pewno przyjdzie.

W pracy pomaga mi mój pies. Na zdjęciu patrzy ma mnie z wyrzutem, że spacer był znów za krótki. Czekam na listy i serdecznie Was pozdrawiam

Wasza Marta







ŚWIATOWY ZJAZD HACKERÓW:

„NIE JESTEŚMY KRYMINALISTAMI"

Przenikanie do pilnie (?) strzeżonych komputerowych banków danych już dawno przestało być tylko sportem, niewinną rozrywką czy pasją paru młodych zapaleńców. Z jednej strony władze wielu państw, zaalarmowane przez wojsko, przedsiębiorców i „poważnych" naukowców, przystąpiły do ścigania i karania „piratów informatycznych". Z drugiej strony sami hackersi zaczęli traktować swe hobby niemal jak misję dziejową, co potwierdził zwołany na początku sierpnia do Amsterdamu pierwszy „Światowy Zjazd Hackersów".

Nie represje, lecz lepsze zabezpieczenia i powstrzymanie zbytniej centralizacji — oto jak należy zapobiegać prawdziwemu piractwu informatycznemu, za którym stoją lub w każdej chwili mogą stanąć gangsterzy i terroryści, szpiedzy wojskowi i przemysłowi. Takie jest zdanie większości spośród dwustu uczestników zjazdu, który zebrał się w Amsterdamie pod hasłem „Święto hackersów naszej galaktyki". Przyjechali głównie „piraci" z Europy Zachodniej i Ameryki Północnej, ale w czasie obrad kontaktowali się — oczywiście za pomocą komputera — także z kolegami z innych części świata, na przykład z Nairobi. Na miejsce spotkania wybrali Amsterdam, bo Holandia to jedno z niewielu już państw zachodnich, w których tego typu działalność nie jest jeszcze zaliczana do przestępstw.

„Nie jesteśmy kryminalistami" — mówił 19-letni Paul Denissen, redaktor holenderskiego czasopisma dla hackersów „Hack Tic". Przenikając do systemów informatycznych wielkich przedsiębiorstw, a nawet takich instytucji jak Pentagon czy NASA, hackersi „chcą dać sygnał tym, którzy źle ochraniają niekiedy supertajne przecież informacje" — wyjaśnił Denissen. A organizator zjazdu, Jan Dietvorst, podkreślał: „Centralizacja informacji jest niebezpieczna. Wystarczy dysponować głową na karku i dość prostym sprzętem, żeby dostać się do dowolnego banku danych, jak to wykazał Klub Komputerowy „Chaos" z Hamburga wnikając do systemu informatycznego NASA".

Ostrzegając przed niedostateczną ochroną wielu banków danych uczestnicy zjazdu wypowiadali się jednak przeciwko monopolowi informacyjnemu. Apelowali na przykład o udostępnianie różnych potrzebnych informacji państwom rozwijającym się, które często nie są w stanie same ich zgromadzić. No i oczywiście dzielili się doświadczeniami i zapoznawali nawzajem z różnymi nowinkami technicznymi ze świata komputerów i ze sposobami uprawiania swego niecodziennego hobby.

Jacek Safuta

Ceny mogą być wyższe ze względu na inflację, a poza tym obowiązuje tzw. „przelicznik dolarowy"

	Giełda Bajtka	Sklep Bajtka	Komis	Pewex	RFN	Baltona	CSH i inne
	tys. zł	tys. zł	tys. zł	$	DM	$	tys. zł
SINCLAIR							
ZX 81	120	120	—	—	—	—	—
ZX Spectrum 48	400	350	550	—	80	—	—
ZX Spectrum +	450	550	700	—	90	—	—
Timex 2048	400	400	400	—	—	—	240
ZX Spectrum 128+	550	—	—	—	—	—	—
ZX Spectrum 128+2	—	—	—	—	140	—	—
ZX Spectrum 128+3	—	—	—	—	280	—	—
drukarka Seikosha GP 50s	—	280	—	—	—	—	—
Interface Kempston	25	20	—	—	10	—	9.3
COMMODORE							
Commodore 64	900	650	710	199	290	155	950
VC 20	200	180	—	—	—	—	—
C 16	300	250	—	—	80	—	—
C 116	350	240	—	—	70	—	—
C Plus 4	450	450	380	—	150	—	—
C 128	1100	1100	—	—	399	—	—
C 128 D	1900	—	—	450	820	—	—
Amiga 500	2300	2900	—	—	899	—	—
Magnetofon 1531	150	170	170	48	30	225	—
Stacja dysków Oceanic	700	700	—	—	320	170	—
Stacja dysków 1571	1000	1000	—	199	460	—	—
Drukarka LCIOC	1300	1300	—	—	260	230	—
ATARI							
Atari 800 XL	600	700	470	—	160	—	—
Atari 65 XE	700	650	520	127	—	—	—
Atari 130 XE	800	850	—	199	220	—	—
Atari 520 ST	2000	2000	—	—	—	—	—
Atari 1040 ST	—	—	—	—	1140	—	—
Magnetofon XC 12	140	140	—	36	40	—	—
Stacja dysków 1050	900	900	750	185	300	—	—
Stacja dysków 520 STM	—	700	—	—	—	—	—
Drukarka 1029	400	650	—	—	—	—	—
AMSTRAD							
Amstrad 464 mono.	900	900	—	—	350	—	—
Amstrad 664 mono.	—	1200	—	—	—	—	—
Amstrad 6128 mono.	—	—	—	—	670	—	—
Amstrad PCW 8256	—	—	—	—	—	—	—
Amstrad PCW 8512	—	—	—	—	—	—	—
Amstrad PCW 9512	—	—	—	—	—	—	—
Stacja dysków do 464	—	—	—	—	380	—	—
SHARP							
Sharp MZ 700	—	360	—	—	—	—	—
Sharp MZ 800	—	420	—	—	—	—	—
Dyskietki 5.25 cala	2	2-2.5	3	1	0.7	11	4-9
Dyskietki 3.5 cala	7.5	7.5-9	7-9	—	5	2.5	9-10
Dyskietki 3 cale	11	—	—	—	6	3	10
Joystick	21	16-26	20-30	5	10	10	14
Monitor Neptun	60	80	—	—	—	—	68

INDYWIDUALNY BANK DANYCH

Witold Poznański, uczeń — 16 lat. Posiada ATARI 65 XE z magnetofonem XC—12 oraz około 100 programów. Pragnie nawiązać kontakt z posiadaczami ATARI w celu wymiany oprogramowania, szczególnie z woj. katowickiego. Adres: Al. 35-lecia 22/43, 32-512 Jaworzno.

Piotr Samulewicz, lat 14. Posiada AMSTRADA 464, prosi o kontakt posiadaczy tego komputera w celu wymiany oprogramowania, literatury i doświadczeń. Adres: 83-110 Tczew, ul. Rutkowskiego 45/c/4.

Wojciech Józefiok, lat 17. Posiada COMMODORE C—128 z magnetofonem 1530 DATASETTE. Oprogramowanie: gry oraz programy muzyczne. Zainteresowania: astronomia, fantastyka oraz elektronika. Nawiąże kontakt w celu wymiany doświadczeń i oprogramowania. Adres: 41-707 Ruda Śl. ul. Kamienna 10.

Edward Brodnicki prosi o kontakt posiadaczy komputera CANON BX—1. Sprawa bardzo ważna! Adres: 42—200 Częstochowa, ul. Grochowskiego 6.

Krzysztof Mąka, posiada ATARI 65 XE, stację dysków 1050 oraz ponad 200 programów. Zainteresowany jest wymianą oprogramowania. Adres: 33-336 Łabowa 137, woj. nowosądeckie.

Dariusz Kołaczek, posiada mikrokomputer SHARP—MZ 700 z wbudowanym magnetofonem. Poszukuje programów oraz literatury na temat mikrokomputera MZ—700. Adres: 52—019 Wrocław, ul. Popielskiego 6/10.

Andrzej Cudowski wraz z synem proszą o kontakt posiadaczy stacji dysków do ATARI FX—551 (dwugłowicowa). Są użytkownikami ATARI 800 XL, magnetofonu XC—11, stacji dysków FX—551 oraz drukarki PANASONIC KX—P10801. Posiadają około 400 gier oraz kilkanaście programów użytkowych (przeważnie do obróbki tekstów i współpracy z drukarką). Adres: 95-200 Pabianice, ul. Reymonta 26/1a.

Paweł Masowa, posiada mikrokomputer ITT 2020 monitor oraz drukarkę i stację dysków. Interesuje się muzyką oraz elektroniką. Pragnie nawiązać kontakt z posiadaczami tego komputera. Adres: 83-406 Wąglikowice, Grzybowski Młyn, woj. gdańskie.

Waldemar Szramka, lat 15. Posiada AMSTRADA 464. Nawiąże kontakt z posiadaczami programów użytkowych i edukacyjnych w celu wymiany. Adres: 64-200 Wolsztyn, ul. Różana 27.

Paweł Zakrzewski, lat 15, uczeń LO i Szkoły Muzycznej II. Posiada mikrokomputer MSX HB—75F HIT BIT SONY, joystick MATT I QUICKSHOT II oraz magnetofon PHILIPS. Pragnie nawiązać kontakt z posiadaczami mikrokomputerów MSX (najchętniej z mieszkającymi w Łodzi), w celu wymiany oprogramowania oraz doświadczeń. Adres: 93-161 Łódź, ul. Naruszewicza 10 m 31.

Dariusz Książek, lat 12. Posiada TIMEXA 2048, magnetofon SVI 767 TP. Proponuje wymianę gier i programów. Adres: 45-056 Opole, ul. Kośnego 34a/6.

Przemysław Woźniak, lat 17. Posiada ATARI 65 XE, magnetofon XC 12 oraz 200 programów, w tym gry i programów użytkowych. Adres: 57-220 Ziębice, ul. Wąska 11/5.

Marcin Bielawski, lat 15. Posiada ATARI 130 XE, stację dysków LDW 2000 i magnetofon XC 12. Pragnie nawiązać kontakt z innymi użytkownikami ATARI w celu wymiany doświadczeń i oprogramowania. Posiada około 150 programów użytkowych i gier. Adres: 04-266 Warszawa, ul. Świetlicka 13.

Adam Wawszczyk, lat 10. Posiada mikrokomputer CPC 6128. Prosi o kontakt użytkowników tego komputera w celu zdobycia oprogramowania oraz literatury. Adres: 44-117 Gliwice, ul. Centaura 4/17.

Piotr Wróbel, lat 15. Posiada ATARI 130 XE, stację dysków LDW 2000, magnetofon XC oraz dwa joysticki. Dysponuje zestawem ok. 250 gier i 50 programów użytkowych. Zainteresowania: muzyka disco, komputery i ich programowanie. Adres: 31-979 Kraków, ul. Klasztorna 8.

Drogi Bajtku!

Na listy Czytelników odpowiada Dominik Falkowski

Czy w programie „Dekoder Transmisji Radiowej" nie ma błędu? Wpisałem go i sprawdziłem kilkakrotnie, a suma kontrolna jest o 2 mniejsza od podanej w listingu. Proszę o rozwianie mojej wątpliwości, gdyż program DTR+ jest dla mnie bardzo cenny.

Andrzej Romanek, Gliwice

Program ten jest wydrukowany bez błędu. Przyczyną kłopotów w uruchomieniu jest piąta liczba w linii 330. Powinna ona wynosić 255, a na skutek niedokładnego wydrukowania najmniej znaczącej piątki, brana jest ona za 253. Po poprawieniu i uruchomieniu programu, powinien działać on bez kłopotów.

Posiadam ZX Spectrum i od kilku dni piszę program, który ma się zajmować księgowością. Napotkałem jednak na przeszkodę: w programie tym potrzebne jest sporo danych, więc zadeklarowałem tablice w następujący sposób:

DIM p$ (255,11) DIM j (255,6)
DIM l$ (255,11) DIM k (255,6)
... ...

W sumie jest 20 tablic tekstowych i sześć liczbowych, a komputer sygnalizuje brak pamięci. W jaki sposób „wepchnąć" jednak te dane do pamięci?

Mariusz Wyszopolski, Sochaczew

Dokonajmy prostego rachunku. Z danych podanych w liście wynika, że potrzebne jest 20∗(255∗16)=81600 bajtów na litery i 6∗(14∗(255∗6))=128520 bajtów na liczby. W sumie daje to 210KB pamięci. Teraz już wiadomo, dlaczego „Spectrumna" odmawia... Tu nie poradziłby sobie ani Commodore, ani nawet Amstrad.

Musi Pan zastanowić się, które z tych danych są naprawdę potrzebne, jak można je upakować, okroić itp. W ostateczności proponuję zakup stacji dysków, np. FDD3000 (test wkrótce w Bajtku), która to, dzięki swej szybkości, potrafi przechowywać i przesyłać naprawdę duże ilości danych.

Czy istnieją na Spectrum rozszerzenia pamięci w formie np. pamięci EPROM lub EEPROM? Jeśli tak, to gdzie można kupić potrzebne części i jaki byłby przypuszczalny koszt takiego rozszerzenia? Jak dowiedzieć się potrzebnego hasła w grze „Puszka Pandory"?

Piotr Smolański, Katowice

Rozszerzenie pamięci w postaci EPROM-u byłoby z założenia bardzo niefunkcjonalne, gdyż musiałoby składać się z programatora pamięci, magistrali łączącej z RAM w komputerze i odpowiednich procedur obsługujących. Dlatego też taki sposób rozszerzenia pamięci nie jest stosowany.

Czynione są natomiast próby, być może już zakończone, podłączenia banku pamięci zwiększającego całkowitą pamięć RAM Spectrum do 80KB. Mikroprocesor Z80 może adresować jedynie 64KB, więc dodatkowa pamięć używana być może jako przełączalny bank lub też jako RAMDISC. Mamy nadzieję, że schemat rozszerzenia pamięci zostanie wkrótce zamieszczony w Bajtku.

Jeszcze jedno zastosowanie pamięci 80KB — uruchamianie gier ze Spectrum 128 z muzyką na generator AY z pewnością znajdzie uznanie wśród graczy.

Co do gry „Puszka Pandory", to hasło zapomniał nawet jej autor, red. Marcin Borkowski. Tak więc grę musi każdy rozpracować sam.

Jakie taśmy używać najlepiej do przechowywania programów? Zaczynam zbierać gry i ma to dla mnie znaczenie.

Tadeusz Stasiak, Zakopane

Programy z taśm będą odtwarzane wielokrotnie. Sama taśma będzie też kasowana, często przewijana, więc musi być prawie niezawodna. Do programów najlepiej stosować taśmy o długości 2×15 minut. Szybki czas dostępu będzie jednak szedł w parze ze zwiększeniem liczby kaset w zbiorze. Dlatego, jeśli masz naprawdę dużo programów, używaj taśm 2×30 minut, o dobrej jakości i dużej dynamice, lecz tylko Normal!

Od kilku miesięcy mam komputer TIMEX 2048 i od tego czasu zaczęły się moje kłopoty. Próbuję wpisywać program z „Bajtka" i inne, dla ZX Spectrum. Ale żaden z tych programów nie chce pracować. Na przykład w książce „Mój mikrokomputer ZX Spectrum" jest prosta gra komputerowa. Linia 210 IF INT Y <>INT A THEN GOTO 250 w żaden sposób nie chce się wprowadzić. Dlaczego?

Kupiłem na giełdzie 4 kasety; w sumie 48 programów. Uruchamia się tylko 11. Niektóre w ogóle nie wchodzą, lub na ekranie pojawia się zamiast tytułu 10 znaków zapytania. Czy można temu zaradzić? I ostatnie pytanie, to prośba o nieśmiertelność do gry „Batty".

Stanisław Trojanowicz, Wadowice

Źródło trudności we wpisywaniu programów jest bardzo prozaiczne. Po prostu robi Pan błędy. Przykładowo linia 210 zawiera znak „<>", który znaczy „jest różne niż". W Spectrum jest to jeden znak, o kodzie 201, a znajduje się na literze W.

Programy kupione na giełdzie, które wieszają się lub nie chcą chodzić, to jeszcze jeden przykład, że nie warto kupować u pasera. Wprawdzie na razie nie ma w Polsce innej możliwości, więc jedynym miejscem dostępu do programów jest giełda. Aby programy uruchomić, musi Pan próbować wgrywać je wielokrotnie, na różnym ustawieniu siły głosu i barwy w magnetofonie. Można również pokręcić nieco wkrętem dociskającym głowicę, regulując na słuch skos głowicy.

Mogę podać oczywiście nieśmiertelność do Batty, w postaci adresu i wartości, którą należy tam wpisać. POKE 47633,0 załatwi nieśmiertelność, lecz wpisanie go jest możliwe tylko w przypadku odbezpieczonej wersji gry. Przy oryginalnej wersji wpisanie nieśmiertelności to zadanie dla prawdziwego hackera.

Mam 12 lat i posiadam tylko klawiaturę komputera Timex 2048. Chciałabym wiedzieć, co jeszcze muszę dokupić, aby w pełni korzystać z komputera.

Marzena Lis, Dębica

Jeżeli w istocie masz tylko klawiaturę, to kupić musisz wszystko. Jeśli jednak słowem „klawiatura" określasz pudełko z klawiszami, gniazdami i wyłącznikiem, to masz tylko komputer.

Jak zapewne wiesz, każdy komputer składa się z jednostki centralnej, konsoli i urządzeń wejścia/wyjścia. Timex 2048 to jednostka centralna (mikroprocesor, pamięci itp.) oraz klawiatura (same klawisze!). Potrzebny jest jeszcze monitor, który zastąpić można telewizorem z systemem PAL i 36 kanałem UHF. Jako urządzenie wejścia/wyjścia, a raczej pamięć masową stosować możesz magnetofon, najlepiej kasetowy. Powinien on posiadać wyjście słuchawkowe „jack" i takie samo wejście mikrofonowe. Jeśli jedno z nich, lub też oba, jest typu DIN, można w prosty sposób wykonać przejściówkę.

Jeśli kupisz jeszcze joystick i pożyczysz kilka gier, to już możesz w pełni korzystać ze swego komputera.

W numerze 6/89 podany był schemat generatora dźwiękowego do Spectrum. Mam zamiar wykonać go, lecz nie umiem zaprojektować płytki. Jeśli jest to możliwe, to proszę o podanie w „Bajtku" płytki potrzebnej do zmontowania generatora, gdyż bardzo mi na tym zależy.

Piotr Woźniak, Kielce

Projekt płytki przygotowywany jest do druku i najprawdopodobniej ukaże się z początkiem 1989 roku, w Klanie Spectrum. Możesz jednak i bez tego zmontować generator, łącząc elementy „w powietrzu", odpowiednio izolując połączenia. Konstrukcja jest stosunkowo prosta i nie powinna sprawiać kłopotów. Pamiętaj, by generator był przelotowy!

UWAGA! OGŁASZAMY WIELKI TURNIEJ!

Wszyscy wiemy, jak potrzebnym urządzeniem jest joystick. Znajduje się on prawie przy każdym komputerze, poczynając na Spectrum a na IBM kończąc.

Żaden monopolista nie zdoła więc nasycić rynku swymi joystickami. Napływają one z zagranicy, sprowadzane przez osoby prywatne, czasem także i firmy. Produkowane są również w kraju. W sumie na rynku znajduje się ok. 15 typów joysticków.

Różnią się one wielkością, trwałością, konstrukcją. Nikt nie wie, który jest najlepszy. Bo i nie ma takiego. Delikatne, analogowe joysticki są niezastąpione w programach graficznych, zaś solidne konstrukcje na mikroprzełącznikach — w najkrwawszych grach. Wreszcie podstawowa różnica — cena automatycznie promuje joysticki najtańsze.

Dlatego też postanowiliśmy ogłosić i przeprowadzić wielki turniej. Prosimy producentów, dystrybutorów i wszystkie instancje związane z produkcją joysticków na Polskę, o zgłoszenia do turnieju.

Każdy ze zgłoszonych joysticków będzie poddawany różnorodnym próbom wytrzymałościowym. Przewidujemy moczenie w Coca-Coli, kręcenie nad głową, próbę dżemu, zginanie, wyrywanie, test przyssawek, wreszcie wciskanie FIRE kilkaset tysięcy razy. Lista konkurencji będzie się zwiększać, w miarę jak napływać będą nowe pomysły.

Wyniki turnieju opublikujemy w „Bajtku" jak najszybciej, zaś zwycięzca otrzyma tytuł Joysticka Roku.

Zgłoszenia prosimy nadsyłać na adres:

Bajtek
ul. Wspólna 61
00-687 Warszawa
„Joystick Roku"





KOMU SZKODZĄ KOMPUTERY?

— *dwunastu studentów próbowało popełnić zbiorowe samobójstwo z radości po osiągnięciu 1 000 000 punktów w ulubionej grze,*

— *11-letni chłopiec zabił kolegę; wydawało mu się, że grają w grę komputerową,*

Od długiego czasu zasypywani jesteśmy lawiną informacji, jakie to korzyści niesie komputeryzacja. Wnioski te wypływają jednak po globalnym rachunku, który nie uwzględnia jednostek i przypadków szczegółowych.

Zacznijmy od samego początku. „Informatyka", „komputeryzacja", czyli po prostu mikroprocesory goszczą u nas w sposób odczuwalny od około 10 lat. Od tego czasu zdążyły zawładnąć prawie każdym urządzeniem, od pralki, aparatu fotograficznego do zwykłej windy osobowej.

Na początku jednak utożsamiane były głównie z automatami „do gier", rozstawionych masowo w lunaparkach, objazdowych budach i podziemnych przejściach. Stoją tam do dzisiaj i grają przy nich wciąż te same osoby, wrzucając w żarłoczne otwory coraz większe nominały.

Nie zdajemy sobie sprawy, jaką plagą mogą być automaty. Spójrzmy na jeden z krajów zachodnich, gdzie, jak się wydaje, komputeryzacja gości od dawna. Tam każdy sposób zdobycia pieniędzy jest dobry. Złotą żyłę kilku przedsiębiorczych panów zauważyło w kieszeniach nastolatków, tych bliżej dolnej granicy lat -nastu.

W rezultacie zagęszczenie automatów na ulicach np. Londynu jest równe w przybliżeniu zagęszczeniu chłopców, którzy mają ochotę zagrać. I robią to niemalże bezustannie. A szpary automatów pochłaniają jedną dwudziestopięciopensówkę za drugą.

Granie staje się narkotykiem. I jak narkotyk nie daje zażywającemu ni chwili wytchnienia, tak i automaty bez przerwy zaprzątają umysły młodych chłopców.

Jakie są tego następstwa? Zdesperowani nastolatkowie podbierają rodzicom pieniądze. Gdy raz i drugi zbiorą za to baty, próbują kraść lub zastawiać w lombardzie wynoszone z domu przedmioty. Czasem kilku graczy łączy się w większą grupkę, która napada kolegów zwabiona ich kieszonkowym. Jeżeli nie zostanie to w porę zauważone i przerwane, może być szkodliwe dla społeczeństwa dużo bardziej, niż możemy to sobie wyobrazić. Prostytucja nieletnich i napady rabunkowe to przykłady standardowe.

Na szczęście sytuacje takie mają miejsce stosunkowo rzadko, lecz mimo to powinny być dzwonkiem ostrzegawczym dla odpowiednich instancji, gdyż młodych ludzi nie wolno bezkarnie deprawować.

Zdarzają się również przypadki ekstremalne, tragiczne w swej wymowie. Oto jeden z nich:

W pewnym angielskim mieście żyło dwóch jedenastoletnich chłopców, dobrych kolegów. Można łatwo domyśleć się, iż byli oni zapalonymi graczami. Szczególnie ulubili sobie grę „Double Dragon", w której walczyli przeciwko sobie, kierując postaciami na ekranie automatu.

Pewnego dnia chłopcy bawili się w ogrodzie. Zabawa przerodziła się w mocowanie i gonitwę. W pewnej chwili jeden z chłopców wziął nóż i wbił drugiemu w brzuch...

Jak się później tłumaczył, wydawało mu się, że grają w ulubioną grę „Double Dragon" i zabicie przeciwnika było najrozsądniejszym wyjściem.

Dlaczego tak się stało? Kto dopuścił do tej tragedii? Na te pytania nie ma odpowiedzi, lecz można zrobić wiele, by podobny przypadek już się nie powtórzył.

Zastanówmy się teraz, jaki jest wpływ komputerów na ludzi nieco starszych i wydawałoby się, rozsądniejszych.

Szwedzki student informatyki został umieszczony w szpitalu psychiatrycznym, zwanym popularnie „domem wariatów", gdyż wydaje mu się, że jest komputerem. Pisze sam sobie programy w Turbo Pascalu i sam je wykonuje. Bliższe szczegóły nie są nam znane, jedynie to, że student ten znajduje się pod ścisłą obserwacją, mimo iż nie jest groźny dla otoczenia.

Dwunastu studentów próbowało popełnić zbiorowe samobójstwo skacząc do morza. Było to wypełnienie ich ślubowania, iż zabiją się, gdy osiągną granicę miliona punktów w grze „Asteroids". I osiągnęli...

Głupieją więc nie tylko dzieci, lecz i poważni młodzi ludzie. W ich przypadku motywów postępowania nie da się łatwo wyjaśnić, niemniej jednak przypadki te są niezwykle rzadkie.

A starsze pokolenie? Czy ono, wychowane bez komputerów potrafi się przed nimi obronić? Znana jest historia pięciu tysięcy drukarzy, których zwolnił z pracy Rupert Murdoch po skomputeryzowaniu poligrafii w Anglii. Cóż mogli oni zrobić? Tylko nieliczna garstka znalazła pracę w zawodzie, większa część zupełnie poza nim. Spora grupa zbiedniała; z pozycji poważanego drukarza przeszli oni do rzędów biedoty, bez nadziei i środków do życia.

Bardzo trudno jest oszukać komputer. Przekonał się o tym pewien pracownik amerykańskiego biura podróży gdy przyłapywany był na najdrobniejszej próbie oszustwa po skomputeryzowaniu biura. W napadzie szału zniszczył całą sieć komputerową i osadzono go w szpitalu psychiatrycznym.

Co można uczynić, by zaradzić złu, którego źródło — jak się wydaje — leży w komputeryzacji? Nie należy jej przede wszystkim degradować. Korzyści są bowiem niepomiernie większe, niż straty. Ale nie to jest najważniejsze.

Najważniejszym stwierdzeniem jest to, że źródło zła leży w człowieku. Potrafi on postępować niegodnie i nagannie w każdej sytuacji, a oprócz tego każdy z nas ma coś, co sprawia, że jest potencjalnym wariatem, mordercą lub złodziejem. Większość na szczęście potrafi poradzić sobie z ciemną stroną charakteru; u niektórych to ona przeważa.

Powraca pytanie: co zrobić? Należy zacząć od dzieci. Uważać, w jakim przebywają środowisku, jakie oglądają filmy. Limitować również ich kontakt z komputerem (chodzi głównie o gry), jeśli nie potrafią robić tego same. Otaczać miłością i dobrocią.

Jeśli chodzi o obywateli trochę starszych, to ważne jest, by mieli co robić. Z nudów przychodzą do głowy pomysły, które mogą skończyć się tragicznie. Zaś człowiek zajęty stwarza dużo mniejsze niebezpieczeństwo.

Choć w prawie każdym przypadku winni są ludzie, na koniec przykład człowieka, który rzucił się z okna. Za dwie miejscowe rozmowy telefoniczne komputer policzył mu bowiem pomyłkowo ponad 4000 $ rachunku.

Marcin Przasnyski

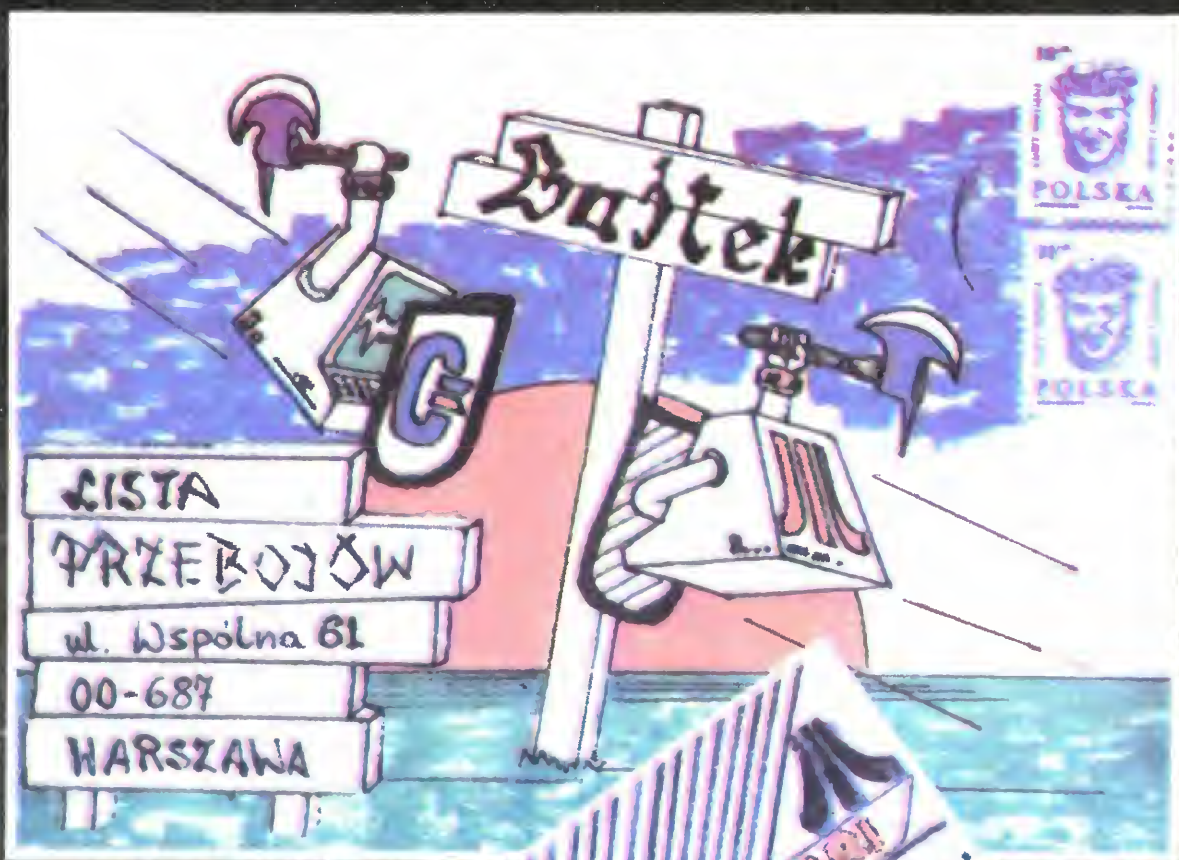
— *pracownik biura podróży oszalał z powodu niemożności oszukiwania po skomputeryzowaniu biura.*

MALOWANE KOPERTY

Najpierw było nam bardzo miło, że autorzy listów do „Bajtka" zadali sobie trochę więcej wysiłku — niż wymaga tego tylko zaadresowanie koperty — i przesyłają nam przy okazji efekty swej twórczości rysunkowej i malarskiej. Ale gdy stojące w redakcji kilka wielkich tekturowych pudeł po monitorach zapełniło się takimi malowanymi kopertami — doszliśmy do wniosku, że jest to nie tylko miły gest Czytelników. Bo faktem jest, że wyrażają oni w ten sposób swoją sympatię w stosunku do „Bajtka". Oczywiste jest też, że ten rodzaj korespondencji jest pośrednim mobilizowaniem redakcji do starania się o utrzymanie jak najwyższego poziomu pisma. Ale jest w tym chyba również coś więcej...

Uznaliśmy, że szkoda byłoby, aby te tysiące przesyłek zmarnowało się. Zbyt dużo serca, fantazji, a czasami również i talentu w nie włożono. Dlatego postanowiliśmy zaprezentować w tym miejscu przynajmniej drobny ułamek będących w naszej dyspozycji zbiorów. Myślimy o zorganizowaniu wystawy, przy okazji jednej z zaplanowanych na najbliższe miesiące w Polsce imprez komputerowych, gdzie te unikalne przesyłki do „Bajtka", przede wszystkim do naszej „Listy przebojów", mogłyby w większej ilości — ku satysfakcji zarówno ich nadawców, jak i widzów — ujrzeć ponownie światło dzienne.

WS



Ludwik Prejs z Olsztyna



Jerzy Pikulinski z Kolna

Karol Igła z Katowic





Sławek Ślusarczy z Zabieczowa



Marek Balcerzak z Kalisza



Wojciech Trzcionka z Cieszyna



ILE JEST NA ZIEMI KOSMODROMÓW?

Przyzwyczailiśmy się, słuchając komunikatów o kolejnych startach statków kosmicznych, do takich nazw jak: Bajkonur i Przylądek Cannaveral. Nie są to jednak jedyne miejsca, z których ludzkość szturmuje kosmos. Kosmodromy znajdują się prawie na wszystkich kontynentach: w Europie, Azji, Ameryce Północnej i Południowej, Afryce i Australii. Nie ma tylko kosmodromów na Antarktydzie, ale i stamtąd startują rakiety meteorologiczne, aby badać stan atmosfery nad południowym biegunem naszej planety.

Kompleksy startowe umożliwiające wysyłanie sztucznych satelitów Ziemi, stacji międzyplanetarnych oraz sond penetrujących górne warstwy atmosfery ma obecnie dziesięć państw: Związek Radziecki, Stany Zjednoczone, Francja, Japonia, Chiny, Indie, Szwecja, Norwegia, Włochy i Australia.

Pierwszym kosmicznym portem naszej planety był Bajkonur. To z niego właśnie wystartował w październiku 1957 roku pierwszy sztuczny sputnik Ziemi, stąd rozpoczynały się pierwsze trasy wiodące na Księżyc, stąd, z okrzykiem „Pojechali!", wystartował do swego historycznego lotu Jurij Gagarin... Trudno wyliczyć wszystkie starty, jakich dokonano w Bajkonurze. Dokonywano ich zarówno realizując radziecki program badań kosmosu, jak i w ramach programów międzynarodowych. Z nowych urządzeń startowych, zbudowanych w stepie Kazachstanu, wystartowała do swych próbnych lotów uniwersalna rakieta „Energia" oraz radziecki prom kosmiczny „Buran".

O 10 lat wcześniej rozpoczęła się historia poligonu rakietowego Kapustin Jar. Status kosmodromu uzyskał on jednak dopiero w roku 1964, gdy z jego urządzeń startowych zaczęto wysyłać na orbitę wokółziemską sputniki serii „Kosmos", a później (od 1969 roku) sputniki przygotowane w ramach programu „Interkosmos". W kronice tego kosmodromu znajdują się również starty rakiet „Wertikal", indyjskich sputników „Aribchata" i „Bachaskara", francuskiego satelity „Snieg-3"...

Trzecim radzieckim kosmodromem jest Plesieck. Radzieccy badacze kosmosu nazywają go powszechnie Północnym. Stąd rozpoczynały się orbity wielu „Kosmosów", „Mołni" i innych aparatów. Budowę Plesiecka rozpoczęto w roku 1960.

Również trzy kosmodromy posiadają Stany Zjednoczone (nazywane są one tam również poligonami doświadczalnymi): Wschodni (na Przylądku Cannaveral), Zachodni (w Vandenderg na wybrzeżu Pacyfiku) i Wollops (na niewielkiej wyspie w pobliżu atlantyckiego wybrzeża USA).

Kourou i Biskarros — tak nazywają się kosmodromy Francji; Pierwszy znajduje się stosunkowo daleko od Paryża, ulokowano go bowiem w Gujanie Francuskiej, na północno-wschodnim wybrzeżu Ameryki Południowej. Drugi natomiast znajduje się w samej Francji.

Obecnie Francuzi wystrzeliwują swoje statki kosmiczne jedynie z kosmodromu Curu, albowiem kosmodrom Ammagir (lub Hammagir), który znajdował się w Afryce, przestał istnieć w 1967 roku, a całe jego wyposażenie przewieziono do znajdującego się na terenie Francji poligonu Bisarros. Badają tam Francuzi swoje rakiety balistyczne (ale nie wypuścili jeszcze stamtąd żadnego statku kosmicznego, dlatego nie uwzględniliśmy tego kosmodromu na mapie).

Dwa kosmodromy posiada Japonia. Jeden z nich znajduje się na wyspie Tanegasima i nosi taką samą nazwę. Drugie znajduje się na wyspie Kiusiu i nosi oficjalną nazwę Utionura. Oba mają stosunkowo niewielkie wymiary, lecz wyróżniają się bogatym wyposażeniem w aparaturę elektroniczną, telemetryczną, radiolokacyjną i optyczną.

Chiny zbudowały aż cztery zespoły startowe dla swoich statków kosmicznych i sond wysokościowych. Pierwszy — kosmodrom Szuang-Czeng-Czu w Ceiuciuani, na północy centralnej części kraju; drugi — niedaleko od Taiuania (prowincja Szansi), trzeci — w Siczanie, mieście na południu prowincji Syczuń, 2 tys. km na południowy-zachód od Pekinu. I wreszcie czwarty — na wyspie Hainań. Jego budowę rozpoczęto niedawno, jest to jeden z niewielu portów kosmicznych naszej planety znajdujących się w pobliżu równika. Z takich baz wprowadzanie statków na orbitę wokółziemską jest najwygodniejsze pod względem energetycznym, gdyż następuje sumowanie się prędkości ruchu obrotowego Ziemi (która na równiku jest największa) z prędkością samej rakiety. Pozwala to w sposób istotny (nawet do 45 proc.) zwiększyć masę użytkową, którą rakieta-nosiciel jest w stanie wynieść na orbitę wokółziemską.

Na jednej z wysp w Zatoce Bengalskiej znajduje się indyjski kosmodrom Sri Harikota. Zbudowano go w roku 1971 i zrekonstruowano w 1977.

Dwa rakietowe poligony doświadczalne znajdują się za kręgiem polarnym. Pierwszy z nich — Annee — należy do Norwegii, a drugi — Kiruna — do Szwecji. Stąd wystrzeliwuje się wysokościowe rakiety meteorologiczne i geofizyczne oraz sputniki. Położenie tego kosmodromu ma swoje zalety, jako że wystrzeliwane stąd obiekty łatwo jest wprowadzić na orbity polarne.

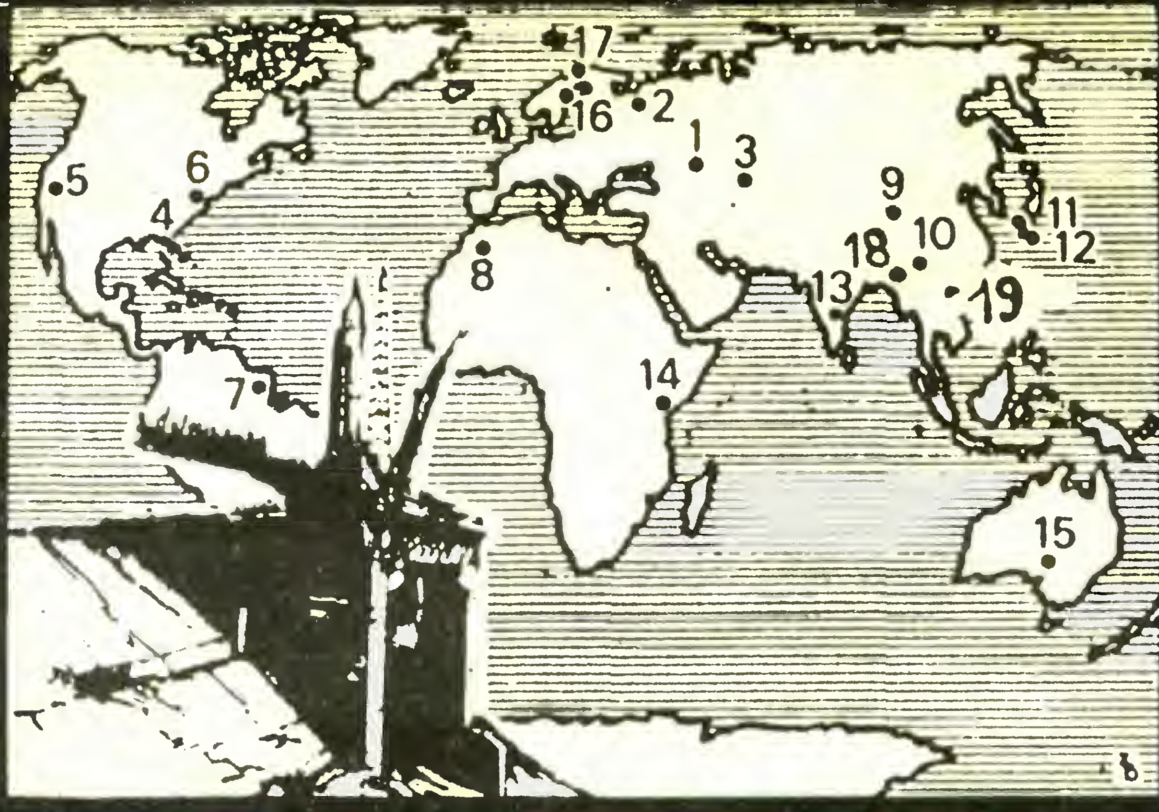
Na południu Australii, obok miasta Wumiera, zbudowano kiedyś doświadczalne centrum rakietowe. Wielka Brytania, po zawarciu umowy z rządem Australii, wystrzeliła stamtąd jeden sputnik i kilka rakiet balistycznych. W roku 1976, po kilku nieudanych startach, postanowiono zamknąć kosmodrom. Jednakże w sierpniu 1988 roku znowu go uruchomiono.

W 1988 roku Australia wystąpiła z inicjatywą zbudowania na jej terytorium (na półwyspie Jork) międzynarodowego portu kosmicznego, który specjalizowałby się w wystrzeliwaniu satelitów na zasadach handlowych. Specjaliści australijscy sądzą, że miejsce wybrane dla przyszłego kosmodromu jest jednym z najlepszych, albowiem znajduje się ono blisko równika, z dala od zamieszkałych punktów i ma świetne warunki klimatyczne.

I wreszcie, włoski kosmodrom San-Marco. Wyróżnia się on tym, że jest obiektem „pływającym". Znajduje się na Oceanie Indyjskim, 5 km od wybrzeża Kenii. Urządzenia startowe składają się z dwóch pływających platform, znajdujących się w odległości 500 m jedna od drugiej i zakotwiczonych za pomocą ruchomych słupów oporowych. Na jednej platformie (San-Marco) zamontowano urządzenia startowe, na drugiej (Santa-Rita) — stanowiska sterowania, łączności i śledzenia lotu.

W sumie jest więc na naszej planecie 19 kosmodromów. Nie wszystkie są wykorzystywane jednakowo intensywnie. Są wśród nich takie, które okazały się czasowo nierentowne (Wumiera). Ale tyle właśnie ich dzisiaj jest. Jednakże tendencja jest taka, że coraz więcej krajów będzie chciało wykorzystywać kosmiczne środki łączności, nawigacji, meteorologii, kartografii, badania bogactw naturalnych itp. Oznacza to, że istniejące kosmodromy będą wykorzystywane coraz intensywniej, a prawdopodobnie — jak twierdzi na łamach „Krasnej Zwiezdy" płk M. Riebrow — rozpoczęta zostanie również budowa następnych.

opr. Waldemar Siwiński



Na rysunku: Schemat rozmieszczenia kosmodromów na kuli ziemskiej: 1. Kapustin Jar (ZSRR); 2. Plesieck (ZSRR); 3. Bajkonur (ZSRR); 4. Przylądek Cannaveral (USA); 5. Vandenberg (USA); 6. Wollopsisland (USA); 7. Kourou (Francja); 8. Ammagir (Francja); 9. Szuang-Czeng-Czu (Chiny); 10. Siczan (Chiny); 11. Utionura (Japonia); 12. Tanegasima (Japonia); 13. Sri Harikota (Indie); 14. San-Marco (Włochy); 15. Wumiera (Australia); 16. Kiruna (Szwecja); 17. Annee (Norwegia); 18. Taiuań (Chiny); 19. Hainań (Chiny).